



Ben-Gurion University of the Negev
Department of Mechanical Engineering

Quadrotor Hovercraft Structure, State Estimation and Control

By: Hanoch Efraim
Supervisor: Dr. Amir Shapiro

Submitted in Partial Fulfilment for the
M.Sc. Degree in Mechanical Engineering

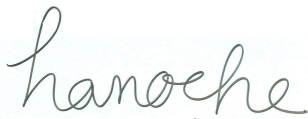
Ben-Gurion University of the Negev
Department of Mechanical Engineering

Quadrotor Hovercraft Structure, State Estimation and Control

By: Hanoch Efraim
Supervisor: Dr. Amir Shapiro

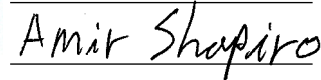
October 2011

Hanoch Efraim:



Date: December 31, 2011

Dr. Amir Shapiro:



Date: December 31, 2011

Prof. Avi Levi :

Date: _____

Submitted in Partial Fulfilment for the
M.Sc. Degree in Mechanical Engineering

Table 0.1: list of symbols

A	world inertial coordinate system, fixed system
$acc_{x,y,z}$	accelerometer measurements
A_d	air density
B	body fixed coordinate system
b	thrust factor
b_i	gyro bias
c	propeller blade chord
C_l	lift coefficient
d	drag coefficient
d_l	velocity momentum coupling coefficient
g	gravity acceleration
I_3	3 by 3 unity matrix
$I_{x,y,z}$	moment of inertia
J	propeller moment of inertia
L	Lyapunov candidate function
m	quadrotor mass
$mag_{x,y,z}$	magnetometer measurements
$p_{1,2,3,4}$	normal to the propellers
R	rotation matrix $R : B \rightarrow A$
R_b	length of propeller blade
V	linear velocity relative to A
$u_{1,2,3,4}$	quadrotor actuators
α	propeller blade angle of attack
β	motor tilt angle
θ	pitch angle
ψ	yaw angle
ϕ	roll angle
ω	hovercraft actual angular velocity
ω_g	gyro sensor reading
$\Omega_{1,2,3,4}$	propeller angular velocities
τ_d	motor time constant

Abstract

In this work we explore the main design and development aspects of a quadrotor hovercraft. These aspects include structure, modeling, control and state estimation. We survey different dynamic models of the system, as well as different stabilization methods. In the field of structure and modeling we will present a new structure of a quadrotor hovercraft with tilted motors. This structure has the desired attribute of natural negative linear velocity feedback. This feedback is useful as it reduces the need for an accurate velocity measurement which can be difficult to achieve. We will validate this attribute using simulations and experiments. In the field of estimation, we will survey the sensors and commonly used sensor fusion methods. We will present a new method for direct measurements of roll and pitch using infrared thermopiles. We will also present a new approach for improving the measurement quality of accelerometers through dynamic vibration damping. In the field of control, we will develop a control law for position regulation utilizing the properties of the tilted rotor structure. It will be shown that no velocity feedback is required for stability. Simulation and experimental results will be presented to validate this theory. The use of angular acceleration feedback in attitude control will be analyzed in the Laplace domain, using 3D plots. We will show that acceleration feedback can improve settling time using pole properties.

Contents

Abstract	IV
List of Figures	VII
Chapter 1. Introduction	11
1.1. Literature Review	13
1.2. Thesis Structure and Contribution	16
Chapter 2. Theoretical Background	19
2.1. Orientation Representation	19
2.2. Kalman Filter	22
2.3. Extended Kalman Filter (EKF)	23
2.4. Complementary Filter	25
2.5. Dynamic Damping	26
2.6. LaSalle's Invariant Set Theory	27
Chapter 3. Structure and Modelling	29
3.1. Classic quadrotor	29
3.2. Quad Tilted Rotor Robot Hovercraft(QTRH)	37
3.3. Aerodynamic Effects	39
3.4. Position Dynamics of the QTRH	44
Chapter 4. Attitude Estimation	45
4.1. Measurements and Sensors	45
4.2. Complementary filter with no bias estimation	53
4.3. Kalman Filter Based Approach	55

CONTENTS	VI
4.4. State Estimation Coupled With System Dynamics	59
4.5. Dynamic Damping	62
Chapter 5. Control	64
5.1. Attitude control analysis based on the linear model	64
5.2. Attitude control analysis based on the non-linear model	71
5.3. Position Regulator	77
Chapter 6. Simulations	83
6.1. Attitude Regulator	83
6.2. Position Regulation	87
6.3. QTRH Simulations	88
Chapter 7. Experiments	91
7.1. Attitude Estimation Experiment	92
7.2. Accelerometers in Attitude estimation	94
7.3. Position Regulator	98
7.4. QTRH Experiments	100
Chapter 8. Conclusions and Future Work	103
Chapter 9. Apendix	107
9.1. Discrete quadrotor model	107
9.2. Attitude estimation based on visual feedback	108
Bibliography	112

List of Figures

2.1 block diagram of a complementary filter	25
2.2 Dynamic damping mechanism	26
3.1 earth and body coordinate Systems	29
3.2 A block diagram of a discrete control system controlling a continuous process	35
3.3 Digital model step response versus continues model step response.	36
3.4 Tilted motor structure	37
3.5 Linear motion at x direction	39
3.6 The propeller angle of attack changes because of linear speed	40
3.7 The propeller angle of attack changes because of linear speed	41
4.1 Accelerometer data along the z axis. In the first 4000 samples electric motors are at rest, and the reading is relatively accurate. At 4000 samples the motors start spinning gradually and the vibrations severely corrupt the data from the sensor.	47
4.2 Thermopile based direct orientation measurement mechanism. The modified servo regulates the thermopiles orientation automatically.	51
4.3 Standard servo motor block diagram	52
4.4 The modified servo motor block diagram	52

4.5 Forces in the z_b, y_b plain in a coordinated turn of an aircraft with mass m	61
4.6 Dynamic Damping mechanism for an Inertial Measurement Unit. 1 - Devise chassis. 2,4,6 - Springs. 3,5,7 - Weights. 8 - Shock absorbent pad. 9 - sensor board	63
5.1 Root locus plot. Simple proportional feedback cannot stabilize the attitude	65
5.2 PD controller block diagram. G is one of the three sub-systems described by 3.11. K_p and K_d are the proportional and differential gains, respectively.	66
5.3 Zero-Pole map. $K_d = 0.3$ while K_p changes between 0.1 to 3	67
5.4 Dominant pole real value as a function of K_p and K_d as computed in simulation	68
5.5 K_p as a function of K_d which gives closest real value for the three poles in simulation	68
5.6 K_p as a function of K_d which gives closest real value for the three poles in simulation	69
5.7 Dominant pole real value as a function of K_p and K_d as computed in simulation	70
5.8 K_p as a function of K_d which gives closest real value for the three poles in simulation	70
5.9 K_p as a function of K_d which gives closest real value for the three poles in simulation	71
6.1 PD Regulator with different k_p gains	84
6.2 PD Regulator roll, pitch and yaw	85

6.3 feedback linearization regulator with different k_p gains, $k_d = 3[\frac{N \cdot m \cdot sec}{rad}]$	86
6.4 feedback linearization regulator roll, pitch and yaw	86
6.7 Unstable position regulation due to small K_d - $K_d = 0.3 \frac{deg \cdot sec}{m}$	89
6.8 QTRH simulation with same gains as in fig.6.7 and $d_l = 0.2[N \cdot sec]$	90
6.9 QTRH simulation with $k_d = 0$ and $d_l = 0.2[N \cdot sec]$	90
7.1 First quadrotor constructed in our lab	91
7.2 Second quadrotor constructed in our lab	92
7.3 IMU evaluation system	93
7.4 IMU evaluation system	94
7.5 IMU evaluation system - pitch measurements and calculations under vibrations.	95
7.6 Raw acceleration data during a near-hover state	95
7.7 Raw acceleration data during a near-hover state	96
7.8 Roll and pitch angles during the flight	97
7.9 Accelerometer data after filtering	97
7.10 pitch from IMU compared with pitch from accelerometers	98
7.11 Position regulator results	99
7.12 Motor tilting Mechanism	100
7.13 Position regulator results with no velocity feedback, $K_p = 7[\frac{deg}{m}]$	101
7.14 Unstable position regulator results - $K_p = 15[\frac{deg}{m}]$, $K_d = 0$	101
7.15 Unstable position regulator results - $K_p = 15[\frac{deg}{m}]$, $K_d = 0$	102

8.1 Variable β QTRH	105
8.2 Variable β QTRH motor mount	106
8.3 Variable β QTRH servo mechanism	106
9.1 Common urban image.	108
9.2 Fig 9.1 two dimensional Discrete Fourier Transform	109
9.3 The same scenery as in 9.1, rotated in 22 degrees	109
9.4 Fig 9.3 two dimensional Discrete Fourier Transform	110
9.5 The same scenery as in 9.1, rotated in 22 degrees	110
9.6 Fig 9.5 two dimensional Discrete Fourier Transform	111

CHAPTER 1

Introduction

The role of robots in our lives is getting bigger and bigger as technology advances. Robots are used in both critical applications, such as medicine procedures and bomb defusal, and at the same time they are used to perform every day tasks, such as cleaning our houses. Technology developments in the field of Micro Electric Mechanical Sensors (MEMS) in the last few years have caused rapid progress in the development of flying robots and Micro Air Vehicles (MAV). These robots have the potential to change many aspects of our lives, with an incredible number of possible uses. Search and rescue, traffic control and real time imaging for security and military are only a small part the possible applications. This work focuses on the issues that arise when developing a quadrotor robot - a Vertical Take Off and Landing (VTOL) robot - its design, state estimation and control.

State Estimation - The state vector of a quadrotor robot is composed of attitude, position and both angular and linear velocities. In older flying machines, which required attitude estimation, a mechanical gyroscope [40] was used. Mechanical gyroscopes are very accurate and provide a direct measurement of attitude, but they are also heavy and expensive. They cannot be used in MAV's due to their size and weight. MEMS gyros are very small and cheap, and they provide an angular velocity measurement. This measurement can be integrated to calculate

attitude. However, they suffer from bias error (small amplitude, slowly changing error), as well as other limitations. MEMS accelerometers and magnetic flux sensors can be used for direct measurement of orientation by measuring the earth's gravity and magnetic fields. One of the problems that arises when using accelerometer for this purpose is that the gravity measured by the MEMS accelerometers is mixed with the quadrotor's accelerations, making the measurement unreliable. These sensors are also very sensitive to vibrations, and this corrupts the high frequency data they produce. In chapter 4 we will describe different methods used to cope with these difficulties.

Low accuracy and low frequency direct position measurements can be produced by a Global Positioning System (GPS) receiver. Air pressure and sonar sensors are used for more accurate and high frequency measurements. Today's GPS receivers can provide data at a rates of up to 4Hz, with an accuracy of about 3 meters. In chapter 5 it will be shown that, for achieving a stable hover for a quadrotor, it is necessary to implement both position and linear velocity feedbacks. The low rate of the GPS receiver combined with its low accuracy make the linear speed estimation extracted from it not accurate enough and complicated to use.

Structure - In chapter 3 we will present a special quadrotor structure that produces mechanical linear velocity feedback, thus eliminating or reducing the need for accurate linear velocity measurement. We will develop a dynamic model that takes into account some aerodynamic effects that are not usually considered.

Control - As apposed to a helicopter, which possesses a complicated mechanical structure which damps the system, the quadrotor is a non stable system and requires a stabilizing controller. In chapter 5 we will discuss different approaches for stabilizing the traditional quadrotor as well as the quadrotor presented in chapter 3. We will discuss both attitude stabilization and position regulation.

1.1. Literature Review

Structure and modelling - One of the biggest advantages quadrotor robot possesses compared to a helicopter is its structural simplicity. Therefor, there is not a lot of work done on the structure itself. A very comprehensive work was done on the subjects of modelling and control by S. Bouabdallah in [6]. Most "classical" quadrotors are composed of four parallel motors at the edges of a cross-like structure, as seen in fig. 3.1. However, some commercially available quadrotors, such as the Mikrokopter [2], use structures with six or eight motors called the Hexakopter and Oktokopter. such structures provide increased reliability and can maintain flight capability even after a motor failure. The Oktokopter might even be able to sustain flight after the failure of two motors. Another unusual design is of the Draganflyer X6 [1], where six motors are mounted on a 'Y' shaped structure. Two motors are mounted on each edge of the 'Y' structure one pointing up and the other pointing down.

One of the reasons for the growing interest in quadrotor robots in the last few years is a result of the advances which took place in the electrical power systems that can be used in MAV's. In most commercially available quadrotors the power source used is a lithium-ion polymer battery pack [9] that has a much better energy to weight ratio than the previously used nickel metal hydride batteries [33]. The electric motors have also improved significantly. A few years ago, brushed DC (Direct Current) [19] electric motors were used as on the "Dragonfly V Ti RC Gyro Stabilized Electric Helicopter" [3], where today's quadrotor robots use BLDC (Brushless DC) motors which are much more efficient, light and cheap. The disadvantage of this technology is the requirement of a complicated microcontroller based driver[20].

Attitude estimation - Attitude estimation is the heart of the system. During this research, we discovered that the flight characteristics are extremely sensitive to changes in the quality of attitude estimation, much more than they are to control gains. A very comprehensive work was done on the subject of Inertial Measurement Unit (IMU) for attitude estimation by Samuel Fux in [13]. This work deals with the algorithms, sensors and implementation of the actual sensor board. In cases where only one dimensional rotation estimation is required, such as in the case of the segway robot [32], there are two common estimation methods used - Kalman filtering [45, 23, 28] and complementary filtering. A comparison between the two methods can be found in [16]. Regarding the three dimensional estimation problem, R. Mahony et al. have published several papers [34, 27, 35] on the design of a complementary filter to combine angular velocity measurement, corrupted

by bias and noise with a direct attitude low frequency measurements, such as, but not limited to, those of accelerometers and magnetometers. However, Kalman filter cannot be used in this problem because of the non linearity of the kinematic equation. The Extended Kalman Filter (EKF) [45], which is a near optimal filter adequate for use in non-linear systems, is used instead [37]. In this approach, the dynamic equation usually used in the prediction step is replaced with the kinematic equation which describes the relations between angular velocity and angular position, using gyro readings as inputs to the equation. The measurement step uses the direct orientation measurement taken from the accelerometers and magnetometers. It is worth noting that adding information from the dynamic model of the physical system can improve results, as seen in [25]. There, the dynamic model of the system is included in an Inertial Navigation System (INS). The estimation is not limited to being based on inertial sensors. The data required for estimation can be acquired using visual feedback[22, 12, 4, 41].

Control - A lot of research has been done in this area. PID methods for stabilization and navigation [44, 5] are very simple and produce good results, however, more complicated technics such as backstepping [24], sliding mode control [11, 43] and feedback linearization [42, 24] have been also researched. Adaptive control [21, 31, 39] that can accommodate to changes in the system, such as structural damage, is also being researched for quadrotor control, but to our knowledge with no publications at this time. In [7] S. Bouabdallah et al. provide a comparison between the use of backstepping and sliding mode controllers for full stabilization of both position and attitude of the quadrotor. In

another work [38] S. Bouabdallah et al. present a comparison between PID and LQ approaches. Some work on control using fuzzy logic approach was done by C. Coza et al. [8]. R. Mahony et al. provides in [36] a coupled estimation and control analysis using Lyapunov methods to prove stability. A work on trajectory generation and tracking for aggressive maneuvers was done by D. Mellinger et al. [29], based on the two dimensional tracking algorithm presented in [14]. Another work on the subject of trajectory tracking was done by M. Hoffmann et al. in [17].

1.2. Thesis Structure and Contribution

This thesis is composed of three main elements - structure and modelling, state estimation, and control. In the first part, the structure and dynamic model of a standard quadrotor will be presented. Then, the Quad Tilted Rotor Hovercraft (QTRH) structure will be presented and its model developed. In the second part, state estimation methods will be discussed, as a preface to the third part in which control strategies for both regular quadrotor and QTRH will be discussed. Finally, the theory will be validated with simulations and experiments.

Structure and modelling-

1 - Quadrotor position regulator requires velocity feedback damping the system. A new structure design will be presented in chapter 3. In this special structure, called the Quad Tilted Rotor Hovercraft (QTRH), the quadrotor motors are tilted towards its symmetry center. This structure has the property of generating negative velocity feedback

mechanically, without a controller intervention, thus eliminating or reducing the need for a fast and accurate velocity measurement, which is difficult to obtain in most practical cases. The dynamic model will be developed, taking into account some aerodynamic effects which are not usually considered.

2 - A discrete model of a classic quadrotor is presented. This model can be very useful for adaptive control and automatic parameter estimation.

Attitude estimation-

1 - A new concept of dynamic vibration damping for improving sensor measurement quality is presented. This concept is very suitable for high vibration applications such as helicopters or any hovercraft powered by an internal combustion motor. The inertial sensors are placed in a structure mounted on the robot using an elastic surface. Three masses are attached with springs to this structure to absorb the energy of vibrations along the three axes.

2 - A new method for direct pitch and roll measurements using infrared thermopile sensors is presented. To overcome the dependency of thermopile measurements on environmental conditions, a mechanism that uses data from the thermopiles to keep them horizontal is being used.

Control-

1 - A control law for position regulation is developed utilizing the properties of the QTRH presented in chapter 3. It is shown that no velocity feedback is required for stability. Simulation and experimental results are presented to validate the theory.

2 -The use of angular acceleration feedback in attitude control is analyzed in the Laplace domain using 3D plots. It is shown that acceleration feedback can improve settling time using pole properties.

CHAPTER 2

Theoretical Background

In order to develop a control law that stabilizes or guides an aircraft, it is first necessary to describe the state of the aircraft relative to earth. Two coordinate reference frames are defined - one attached to the earth (this system will be referred to as system A) and the other is attached to the aircraft (system B). The orientation of system A relative to system B can be described by several methods including quaternion [26] and rotation matrixes [30]. In this work we chose to use rotation matrixes, as was done in [34].

2.1. Orientation Representation

Rotation matrixes. Let x_{ab}, y_{ab}, z_{ab} be the coordinates of the principal axes of B relative to A . The rotation matrix which describes the rotation of B relative to A is defined by:

$$R_{ab} = \begin{bmatrix} x_{ab} & y_{ab} & z_{ab} \end{bmatrix} \quad (2.1)$$

Since the columns of R are mutually orthonormal, it follows that:

$$R_{ab}R_{ab}^T = R_{ab}^TR_{ab} = I_3 \quad (2.2)$$

and therefore:

$$R_{ab}^T = R_{ab}^{-1} \quad (2.3)$$

A rotation matrix may serve as a transformation, taking coordinates of a point from one frame to another. Consider a random point q in space. Let $q_b = (x_b, y_b, z_b)$ be the coordinates of q relative to frame B , and $q_a = (x_a, y_a, z_a)$ the coordinates of q relative to frame A . The relation between q_a and q_b is given by :

$$q_a = R_{ab}q_b \quad (2.4)$$

The action of a rotation matrix on a point can be used to define the action of the rotation matrix on a vector. Let v_b be a vector in the frame B and v_a be a vector in the frame A . The relations between v_a and v_b are:

$$v_a = R_{ab}v_b \quad (2.5)$$

Rotation matrices can be combined to form new rotation matrices using matrix multiplication. If a frame C has orientation R_{bc} relative to a frame B , and B has orientation R_{ab} relative to another frame A , then the orientation of C relative to A is given by

$$R_{ac} = R_{ab}R_{bc} \quad (2.6)$$

Euler angles. Any orientation of the body frame can be achieved by three consecutive elementary rotations around one of its principle

axes. The three angles associated with these three rotations are called the Euler angles. If the order of rotations is as follows - first around Z_b then around Y_b and finally around X_b , and the magnitudes of the rotations are ψ, θ, ϕ respectively, then ψ is called the yaw angle θ is called the pitch angle and ϕ is called the roll angle. The rotation matrices for each of the rotations are:

$$\left\{ \begin{array}{l} R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi) & c(\phi) \end{bmatrix} \\ R_y(\theta) = \begin{bmatrix} c(\theta) & 0 & s(\theta) \\ 0 & 1 & 0 \\ -s(\theta) & 0 & c(\theta) \end{bmatrix} \\ R_z(\psi) = \begin{bmatrix} c(\psi) & -s(\psi) & 0 \\ s(\psi) & c(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{array} \right. \quad (2.7)$$

From here on, $c(\psi)$ will denote $\cos(\psi)$, $s(\psi)$ will denote $\sin(\psi)$, and the same for θ and ϕ .

The body rotation matrix gets the form:

$$R_{ab} = R_z(\psi)R_y(\theta)R_x(\phi) = \begin{bmatrix} c\theta c\psi & -c\phi s\psi + s\phi s\theta c\psi & s\phi s\psi + c\phi s\theta c\psi \\ c\theta s\psi & c\phi c\psi + s\phi s\theta s\psi & -s\phi c\psi + c\phi s\theta s\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \quad (2.8)$$

The roll pitch and yaw angles can be extracted from the rotation matrix as following:

$$\begin{aligned}\theta &= \text{asin}(-r_{31}) \\ \phi &= \text{atan2}(r_{32}, r_{33}) \\ \psi &= \text{atan2}(r_{21}, r_{11})\end{aligned}\tag{2.9}$$

Here $r_{i,j}$ is the element of R which is located at row i and column j . Throughout this work we will use the rotation matrix of a robot to transform vectors from the earth coordinate system to the body coordinate system according to:

$$\begin{aligned}R &: B \rightarrow A \\ R^T &: A \rightarrow B\end{aligned}\tag{2.10}$$

2.2. Kalman Filter

Kalman filter is an optimal observer under many optimization criterions for linear systems with process and measurement uncertainties. The algorithm consists of two main steps - prediction and correction (or update). The prediction step uses a set of equations that describes the system behaviour, usually the dynamic equations, for predicting the state of the system in the next time step. In the correction step, a measurement of the output of the system is used to correct the prediction. The state estimate is computed using a weighted mean of the predicted state and the measured state, where the Kalman algorithm computes the optimal weights. The gains are optimal under the

assumption of zero mean gaussian noise. For the system:

$$x(k) = Ax(k-1) + Bu(k-1) + w(k-1)$$

$$y(k) = Cx(k-1) + v(k-1)$$

where w and v represent the process and measurement noise with covariance matrices $Q(k)$ and $R(k)$, The Kalman prediction is:

$$\hat{x}(k|k-1) = A\hat{x}(k-1|k-1) + Bu(k-1)$$

$$P(k|k-1) = AP(k-1|k-1)A^T + Q(k-1)$$

$\hat{x}(k|k-1)$ is the a-priori state estimate, $P(k|k-1)$ is the a-priori estimation covariance and $\hat{x}(k-1|k-1)$ is the previous Kalman state estimate. The Kalman correction step:

$$K(k) = P(k|k-1)C^T(CP(k|k-1)C^T + R(k-1))^{-1}$$

$$\hat{x}(k|k) = \hat{x}(k|k-1) + K(k)(y(k) - C\hat{x}(k|k-1))$$

$$P(k|k) = (I - K(k)C)P(k|k-1)$$

$\hat{x}(k|k)$ is the Kalman estimate at time step k and $P(k|k)$ is the error covariance of the estimate.

2.3. Extended Kalman Filter (EKF)

The Kalman filter presented in 2.2 is designed for linear processes that are governed by a linear stochastic difference equations. It is not suitable for use in the case of a nonlinear processes:

$$x(k) = f(x(k-1), u(k-1), w(k-1))$$

$$y(k) = c(x(k-1), v(k-1))$$

where f and c are nonlinear functions. A Kalman filter that linearizes about the current mean and covariance is referred to as an extended Kalman filter. As in the regular Kalman filter, the algorithm consists of the prediction step:

$$\begin{aligned}\hat{x}(k|k-1) &= f(\hat{x}(k-1|k-1), u(k-1), 0) \\ P(k|k-1) &= A_k P(k-1|k-1) A_k^T + W_k Q(k-1) W_k^T\end{aligned}$$

Here A is the Jacobian matrix of partial derivatives of f with respect to x , and W is the Jacobian matrix of partial derivatives of f with respect to w :

$$\begin{aligned}A_k[i, j] &= \frac{\partial f[i]}{\partial x[j]}(\hat{x}(k-1|k-1), u(k-1), 0) \\ W_k[i, j] &= \frac{\partial f[i]}{\partial w[j]}(\hat{x}(k-1|k-1), u(k-1), 0)\end{aligned}$$

The update step:

$$\begin{aligned}K(k) &= P(k|k-1) C^T (C P(k|k-1) C^T + V_k R(k-1) V_k^T)^{-1} \\ \hat{x}(k|k) &= \hat{x}(k|k-1) + K(k)(y(k) - c(\hat{x}(k|k-1), 0)) \\ P(k|k) &= (I - K(k) C_k) P(k|k-1)\end{aligned}$$

Here C is the Jacobian matrix of partial derivatives of c with respect to x and V is the Jacobian matrix of partial derivatives of c with respect to v :

$$\begin{aligned}C_k[i, j] &= \frac{\partial c[i]}{\partial x[j]}(\hat{x}(k-1|k-1), 0) \\ V_k[i, j] &= \frac{\partial c[i]}{\partial v[j]}(\hat{x}(k-1|k-1), 0)\end{aligned}$$

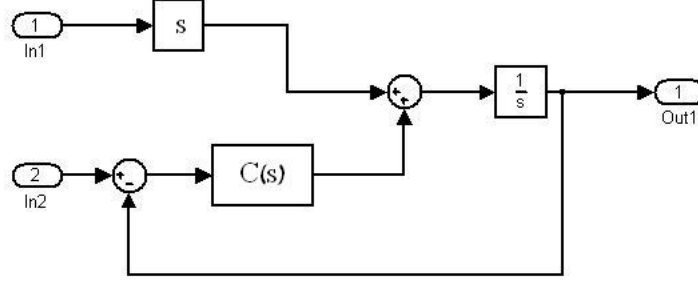


Figure 2.1: block diagram of a complementary filter

2.4. Complementary Filter

The basic idea behind the complementary filter is the combination of a low pass filter and a high pass filter designed in such a way that the sum of their gains is equal to unity for any frequency. The complementary filter is a very useful tool for sensor fusion, where data from one sensor is reliable only at low frequencies and the other only at high frequencies.

Figure 2.1 presents a proposed block diagram for the complementary filter. The relation between $out1$ to $In1$ and $In2$ is given by:

$$\begin{aligned}
 Out1 &= \frac{C(s)}{s + C(s)} In2 + \frac{s}{s + C(s)} In1 \\
 &= T(s) In2 + S(s) In1
 \end{aligned} \tag{2.11}$$

It is easy to see that $T(s) + S(s) = 1$ for any s . $S(s)$ is the sensitivity function of the closed-loop system and $T(s)$ is the complementary sensitivity. For example, If we choose $C(s) = k$ where k is a constant,

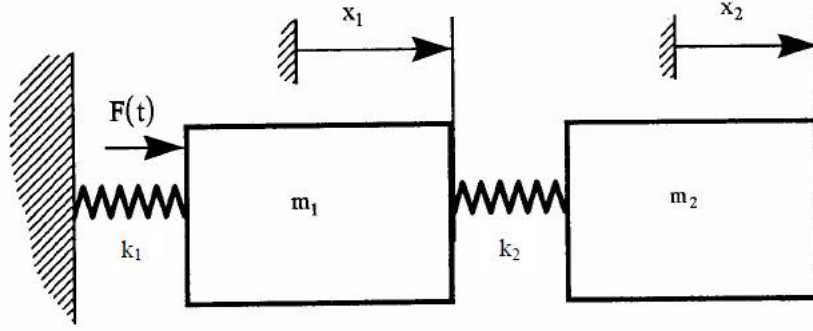


Figure 2.2: Dynamic damping mechanism

$C(s)$ becomes a first order LPF and $S(s)$ a first order HPF. In practice, the differentiator of $In1$ input is not implemented since the data needed to fuse will consist of a direct measurement of attitude and a measurement of angular velocities.

2.5. Dynamic Damping

Figure 2.2 presents a dynamic damping mechanism. The main mass, m_1 represents the object the vibration of which we want to reduce. x_1 and x_2 are the coordinates of m_1 and m_2 . $F(t)$ is the force acting on this mass causing the vibrations. It is assumed that:

$$F(t) = P \cos(\omega t)$$

An additional spring and mass is attached to the primary system in order to absorb the vibrations. k_1 and k_2 are the force constants of the springs. k_2 is chosen such that $k_2 \ll k_1$. m_2 , the attached mass, is

chosen such that $m_2 \ll m_1$. The sum of forces acting on m_1 is:

$$F_1 = F - k_1 x_1 + k_2(x_2 - x_1)$$

The sum of forces acting on m_2 is:

$$F_2 = -k_2(x_2 - x_1)$$

The equation of motion is thus given by:

$$m_1 \ddot{x}_1 + k_1 x_1 - K_2(x_2 - x_1) = P \cos(\omega t)$$

$$m_2 \ddot{x}_2 + K_2(x_2 - x_1) = 0$$

The solution of these equations for x_1 yields:

$$x_1(t) = \frac{k_2 - \omega^2 m_2}{D} P \cos(\omega t)$$

$$D = (k_2 - m_2 \omega^2)(k_1 + k_2 - m_1 \omega^2) - k_2^2$$

This means that if we choose k_2 and m_2 so that $\frac{k_2}{m_2} = \omega^2$, then the amplitude of the oscillations of m_1 will be zero. Note that the resonance frequency of the subsystem k_2, m_2 is $\sqrt{\frac{k_2}{m_2}}$, so in other words, the requirement from the dynamic damping mechanism is that its resonance frequency be as close as possible to the main excitation frequency.

2.6. LaSalle's Invariant Set Theory

Invariant set. A set S is an invariant set for a dynamic system $\dot{x} = f(x)$ if every trajectory $x(t)$, which starts from a point in S , remains in S for all time.

Local Invariant set theorem. Given an autonomous system of the form $\dot{x} = f(x)$, with f continuous and $V(x) : R^n \rightarrow R$ be a scalar function with continuous first partial derivatives. Assume that for some $l > 0$, the set Ω_l defined by $V(x) \leq l$ is bounded and that $\dot{V}(x) \leq 0$ for all x in Ω_l . Let R be the set of all points within Ω_l where $\dot{V}(x) = 0$ and M be the largest invariant set in R . Then, every solution $x(t)$ originating in Ω_l tends to M as $t \rightarrow \infty$.

LaSalle's principle to establish asymptotic stability. If $V(x)$ is a Lyapunov function, and R contains no other trajectories other than $x = 0$, then the origin is asymptotically stable.

CHAPTER 3

Structure and Modelling

The structures of a classic quadrotor and a QTRH are very similar, however, the QTRH structure creates some desired and undesired coupling between state variables. In this chapter the structures and modeling of the quadrotor and of the QTRH are presented.

3.1. Classic quadrotor

The classical quadrotor robot is constructed of four motors placed symmetrically on top of two perpendicular rods, as shown in figure 3.1.

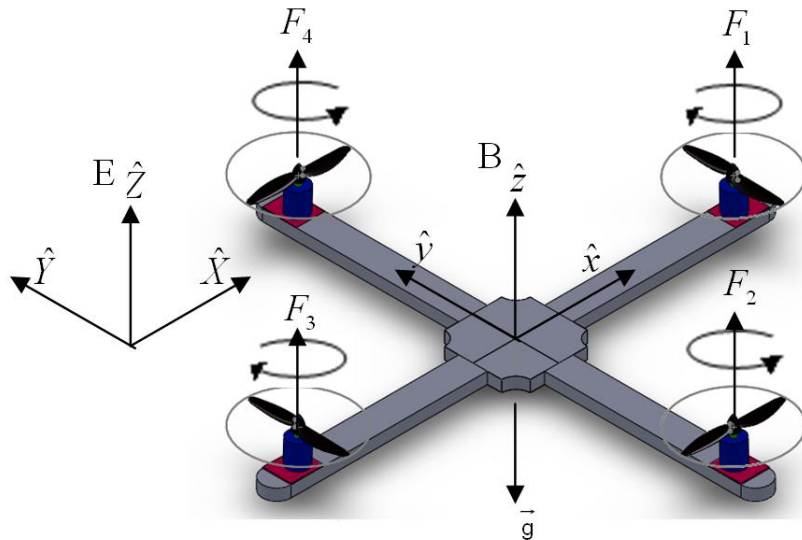


Figure 3.1: earth and body coordinate Systems

It is assumed that the center of gravity is located at the symmetry center, and that the rotation axis of the motors are parallel to each other and perpendicular to the rods. As a result of the symmetry, the inertia matrix I is diagonal, with $I_{x,y,z}$ the moment of inertia around axis x, y, z respectively. There are several different methods for deriving the dynamic model, and different models used in the literature, depending on the approximations made. In ref.[6], the dynamic model is developed using both Newton-Euler Formalism and Euler-Lagrange Formalism to get the following dynamic equations, which describe the hovercraft attitude:

$$\begin{cases} I_x \ddot{\phi} = \underbrace{\dot{\theta}\dot{\psi}(I_y - I_z)}_{A_1} + \tau_x - \underbrace{J\dot{\theta}\Omega_r}_{B_1} \\ I_y \ddot{\theta} = \underbrace{\dot{\phi}\dot{\psi}(I_z - I_x)}_{A_2} + \tau_y + \underbrace{J\dot{\phi}\Omega_r}_{B_2} \\ I_z \ddot{\psi} = \underbrace{\dot{\phi}\dot{\psi}(I_x - I_y)}_{A_3} + \tau_z \end{cases} \quad (3.1)$$

Here $\Omega_r = \Omega_2 + \Omega_4 - \Omega_1 - \Omega_3$ and $\Omega_{1,2,3,4}$ are the four propeller rotation speeds according to fig. 3.1. $\tau_{x,y,z}$ are external moments acting at the body's principal axes by the actuators. The variables ϕ, θ and ψ represent roll pitch and yaw angles, respectively. It is important to note that this model assumes small roll pitch and yaw angles.

These equations can be disassembled into two elements. If one neglects elements $B_{1,2}$ which represent the gyroscopic effect of the propellers under the rotation of the body, the remaining equation is close

to the classic angular momentum equation of a rigid body:

$$I\dot{\omega} = -\omega \times I\omega + \tau \quad (3.2)$$

In the case of a quadrotor, where I is diagonal, eq. 3.2 becomes:

$$\begin{bmatrix} I_x \dot{\omega}_x \\ I_y \dot{\omega}_y \\ I_z \dot{\omega}_z \end{bmatrix} = - \begin{bmatrix} \omega_y \omega_z (I_z - I_y) \\ \omega_z \omega_x (I_x - I_z) \\ \omega_y \omega_x (I_y - I_x) \end{bmatrix} + \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \quad (3.3)$$

The difference between eq.3.1 and eq.3.3, apart from elements $B_{1,2}$, is that in eq.3.3 all quantities are measured in the body frame of reference, while in eq.3.1 angles and velocities are given in the world frame of reference, and $I_{x,y,z}$ are measured in the body frame of reference. For eq.3.1 to be precise, the moments of inertia needs to be measured also in the world reference system. This means that matrix I will change as the quadrotor rotates in space. However, if we assume small angles we get $I_{[A]} \approx I_{[B]}$. The advantage of eq.3.1 is that working with roll, pitch and yaw is more intuitive, where the advantage of 3.3 is that it is more accurate.

In normal flight conditions Ω_r is very small and the effect of $B_{1,2}$ is negligible. Since eq. 3.1 is an approximation for small angles, meaning small velocities for small periods of time, the role of B_1 and B_2 in the model is mainly to note that this effect exist.

The dynamic equations which describe X, Y, Z motion as a consequence of a pitch or roll rotation are often described [7] as:

$$\begin{cases} \ddot{Z} = -g + (c\phi c\theta) \frac{b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2)}{m} \\ \ddot{X} = (c\phi s\theta c\psi + s\phi s\psi) \frac{b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2)}{m} \\ \ddot{Y} = (c\phi s\theta s\psi - s\phi c\psi) \frac{b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2)}{m} \end{cases} \quad (3.4)$$

Here b is the thrust factor and g represents gravity acceleration. The origin of 3.4 is from assuming only two forces act on the robot - gravity and the sum force of the actuators, which acts along the body z_b axis :

$$\begin{bmatrix} \ddot{X} \\ \ddot{Y} \\ \ddot{Z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} + R \begin{bmatrix} 0 \\ 0 \\ \frac{u_1}{m} \end{bmatrix} \quad (3.5)$$

Here R is the rotation matrix. u_1 is the sum of the forces created by the four propellers, and will be defined more precisely in the next section.

Actuators - Control of the robot is done by controlling the four rotor speeds - $\Omega_{1,2,3,4}$. However, it is more convenient to define the

inputs to the system as:

$$\begin{aligned}
 u_1 &= b(\Omega_1^2 + \Omega_2^2 + \Omega_1^3 + \Omega_4^2) \\
 u_2 &= b(\Omega_4^2 - \Omega_2^2) \\
 u_3 &= b(\Omega_3^2 - \Omega_1^2) \\
 u_4 &= d(\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2)
 \end{aligned} \tag{3.6}$$

Here d is the propeller drag coefficient. Substituting 3.6 into 3.1 and 3.4 we get:

$$\begin{cases}
 I_x \ddot{\phi} = \dot{\theta} \dot{\psi} (I_y - I_z) + u_2 - J \dot{\theta} \Omega_r \\
 I_y \ddot{\theta} = \dot{\phi} \dot{\psi} (I_z - I_x) + u_3 + J \dot{\phi} \Omega_r \\
 I_z \ddot{\psi} = \dot{\phi} \dot{\theta} (I_x - I_y) + u_4 \\
 \ddot{Z} = -g + (c\phi c\theta) \frac{u_1}{m} \\
 \ddot{X} = (c\phi s\theta c\psi + s\phi s\psi) \frac{u_1}{m} \\
 \ddot{Y} = (c\phi s\theta s\psi - s\phi c\psi) \frac{u_1}{m}
 \end{cases} \tag{3.7}$$

The control signals $(u_{1,2,3,4})$ determined by the control law are not perfectly translated to actual rotor speeds, because of the dynamics of the electric motors and propellers. The dynamics of the power system is modeled while taking into consideration motor mechanical and electrical parameters at [6]. We have chosen to model it as a first order delay, as done in [17]. This approach was also verified by experiments.

The dynamic behaviour of the actuators in the time domain is thus described by:

$$u_i = u_{iref} * e^{-\tau_d t} \quad (3.8)$$

Here the ' $*$ ' represents the convolution operator, τ_d is the time constant and u_{iref} is the control signal calculated by the controller. In the Laplace domain we get:

$$U_i = \frac{1}{(s\tau_d + 1)} U_{iref} \quad (3.9)$$

Linear approximation of attitude dynamics - It is useful to study the linear approximation of the quadrotor model produced by neglecting the non linear parts. This is equivalent to the assumptions of small angular velocitys and close propeller speeds ($\Omega_1 \approx \Omega_2 \approx \Omega_3 \approx \Omega_4$).

$$\begin{cases} \ddot{\phi} = \frac{l}{I_x} u_2 \\ \ddot{\theta} = \frac{l}{I_y} u_3 \\ \ddot{\psi} = \frac{1}{I_z} u_4 \end{cases} \quad (3.10)$$

In the Laplace domain, and after substituting 3.9 into 3.10 we get :

$$\begin{cases} \Phi = \frac{l}{I_x} \frac{1}{(\tau_d s + 1)} \frac{1}{s^2} U_{2ref} \\ \Theta = \frac{l}{I_y} \frac{1}{(\tau_d s + 1)} \frac{1}{s^2} U_{3ref} \\ \Psi = \frac{1}{I_z} \frac{1}{(\tau_d s + 1)} \frac{1}{s^2} U_{4ref} \end{cases} \quad (3.11)$$

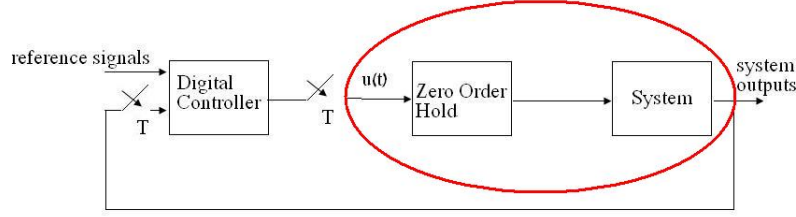


Figure 3.2: A block diagram of a discrete control system controlling a continuous process

In chapter 5 we will use this model to conduct a linear analysis of different control laws.

Discrete time Linear model - The quadrotor itself is a physical system, which exist under a continuous time line. However, the controller is usually implemented using a microprocessor, which is discrete in nature. For this reason it is useful, for stability analysis and parameters estimation, to study the discrete model of the system. As seen in fig.3.2. The physical system is preceded by a zero order block. It's transfer function is:

$$H_{ZOH} = \frac{1 - e^{-sT}}{s} \quad (3.12)$$

Where T is the sampling time, or the control loop interval. The overall transfer function of the Zero Order Hold and the system itself is given by multiplying 3.12 and 3.11. After some calculations (see 9.1) and using Laplace to Z transform conversion tables ([10] appendix 5), we

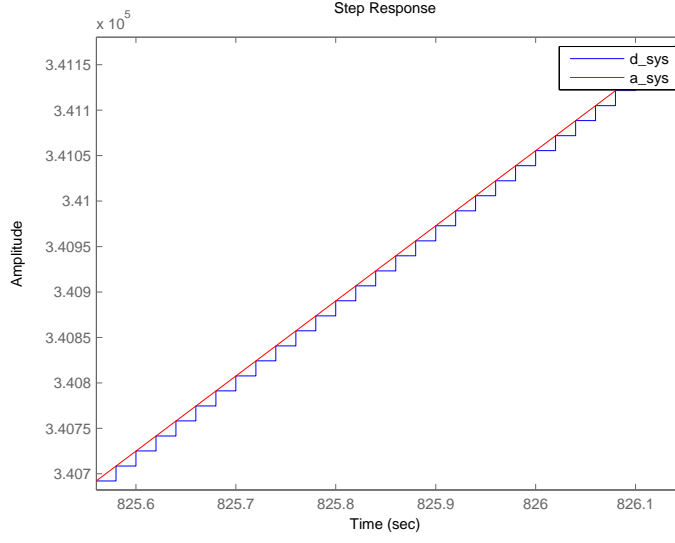


Figure 3.3: Digital model step response versus continues model step response.

get:

$$\left\{ \begin{array}{l} \Phi(z) = \frac{l}{2I_x} \frac{Az^2+Bz+C}{-z^3+Dz^2+Ez+F} U_{2ref}(z) \\ A = -2\tau_d^2 + 2T\tau_d - T^2 + 2\tau_d^2 e^{-T/\tau_d} \\ B = -4\tau_d^2 e^{-T/\tau_d} + 4\tau_d^2 - 2T\tau_d e^{-T/\tau_d} - 2T\tau_d + T^2 e^{-T/\tau_d} - T^2 \\ C = -2\tau_d^2 + 2\tau_d^2 e^{-T/\tau_d} + 2T\tau_d^2 e^{-T/\tau_d} + T^2 e^{-T/\tau_d} \\ D = 2 + e^{-T/\tau_d} \\ E = -1 - 2e^{-T/\tau_d} \\ f = e^{-T/\tau_d} \end{array} \right. \quad (3.13)$$

The same applies for the pitch and yaw angles. Equivalency of the discrete model and continuous model was verified by simulation as seen in fig. 3.3

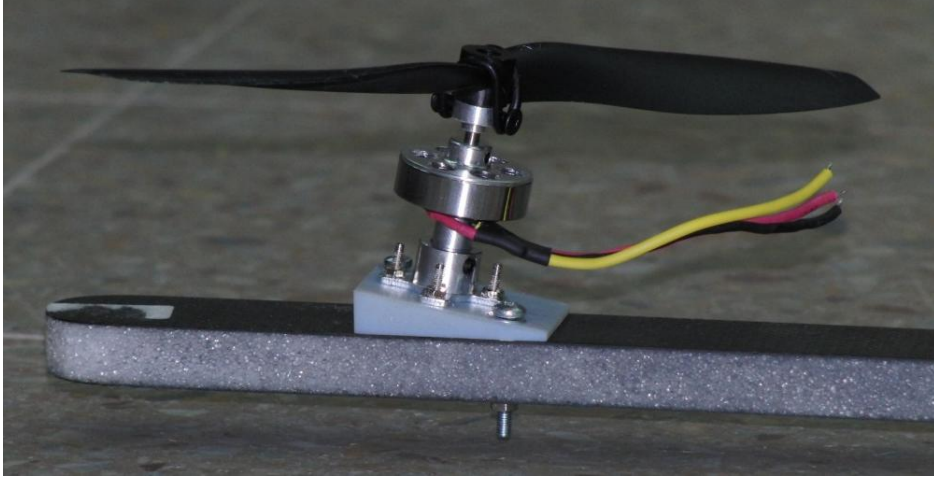


Figure 3.4: Tilted motor structure

3.2. Quad Tilted Rotor Robot Hovercraft(QTRH)

The structure we propose is identical to the structure of the classical structure with one difference. In this structure the motors are tilted towards the center of the robot as depicted in figure 3.4. This structure results in a slightly different dynamic model which will be presented here. In the new model we will also take into account some desirable aerodynamic effects. We will start by calculating the nonconservative torques acting on the body and then assign them in 3.2. We used the same modeling methodology used by [6]. The nonconservative torques acting on the quadrotor as a result of the thrust difference of each pair:

$$\begin{cases} \tau_{xa} = c\beta bl(\Omega_4^2 - \Omega_2^2) \\ \tau_{ya} = c\beta bl(\Omega_3^2 - \Omega_1^2) \end{cases} \quad (3.14)$$

Here β is the tilt angle of the motors. The nonconservative torques acting on the quadrotor as a result of the drag difference between the four rotors:

$$\begin{cases} \tau_{xd} = s\beta d(\Omega_2^2 - \Omega_4^2) \\ \tau_{yd} = s\beta d(\Omega_1^2 - \Omega_3^2) \\ \tau_{zd} = c\beta d(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \end{cases} \quad (3.15)$$

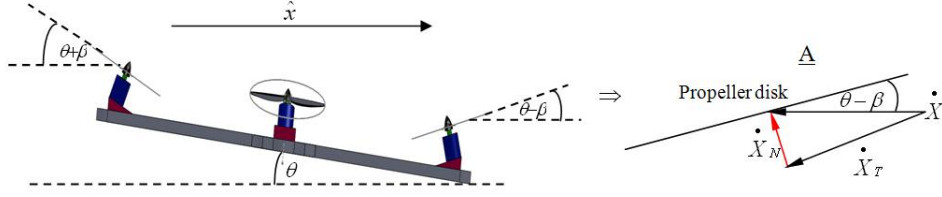
The nonconservative torques acting on the quadrotor as a result of gyroscopic effect from the propellers rotation:

$$\begin{cases} \tau_{xg} = -c\beta J_r \dot{\phi}(\Omega_1 - \Omega_2 + \Omega_3 - \Omega_4) + s\beta J_r \dot{\psi}(\Omega_4 - \Omega_2) \\ \tau_{yg} = c\beta J_r \dot{\theta}(\Omega_1 - \Omega_2 + \Omega_3 - \Omega_4) - s\beta J_r \dot{\psi}(\Omega_3 - \Omega_1) \\ \tau_{zg} = s\beta J_r \dot{\phi}(\Omega_2 - \Omega_4) + s\beta J_r \dot{\theta}(\Omega_3 - \Omega_1) \end{cases} \quad (3.16)$$

And the total nonconservative torque:

$$\begin{cases} \tau_x = \tau_{xa} + \tau_{xd} + \tau_{xg} \\ \tau_y = \tau_{ya} + \tau_{yd} + \tau_{yg} \\ \tau_z = \tau_{za} + \tau_{zd} + \tau_{zg} \end{cases} \quad (3.17)$$

The quadrotor dynamic model describing the roll, pitch and yaw rotations, when not taking into consideration the aerodynamic effects

Figure 3.5: Linear motion at x direction

caused by linear motion, is then:

$$\begin{cases} I_x \ddot{\phi} = \dot{\theta} \dot{\psi} (I_y - I_z) + \tau_x \\ I_y \ddot{\theta} = \dot{\phi} \dot{\psi} (I_z - I_x) + \tau_y \\ I_z \ddot{\psi} = \dot{\phi} \dot{\theta} (I_x - I_y) + \tau_z \end{cases} \quad (3.18)$$

3.3. Aerodynamic Effects

In this section we will consider the case where the quadrotor has a linear speed $V = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix}$ and the aerodynamic effects this speed will have on the tilted rotors. We will begin by analyzing the case where the quadrotor moves at the direction of x , and also that y is parallel to Y , as seen in fig 3.5. We assume that there is no wind, thus the speed V is also the speed of the center of mass relative to the air.

In this case the speed of the imaginary disk created by propellers 1 and 3 relative to the air gets a perpendicular component \dot{X}_N . This causes a change in the angle of attack of the propeller blades as illustrated in fig 3.6

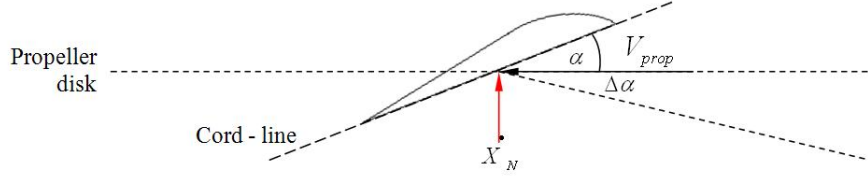


Figure 3.6: The propeller angle of attack changes because of linear speed

Propeller lift:

$$L = \int \frac{1}{2} A_d v^2(r) C_l(r) c(r) dr \quad (3.19)$$

Here A_d is the air density, $v(r)$ is the blade speed at radius r , $c(r)$ is the blade chord and $C_l(r)$ is the lift coefficient. C_l changes with the angle of attack as a non linear function and is dependent on the blade shapes.

A typical plot of C_l as a function of attack angle is shown in fig 3.7.

We will use the following approximation:

$$\frac{dC_l}{d\alpha} = const \triangleq M_{cl} \quad (3.20)$$

And therefore:

$$\Delta L = \int \frac{1}{2} A_d v(r)^2 \Delta C_l c(r) dr = \int \frac{1}{2} A_d v(r)^2 c(r) M_{cl} \Delta \alpha dr \quad (3.21)$$

Using simple trigonometric calculus we get:

$$\begin{cases} \Delta \alpha(r) = \arctan\left(\frac{\dot{X}_N}{V_{prop}(r)}\right) = \arctan\left(\frac{\dot{X}_N}{\Omega r}\right) \\ v^2(r) = (\dot{X}_N)^2 + V_{prop}^2 = \dot{X}_N^2 + (\Omega r)^2 \end{cases} \quad (3.22)$$

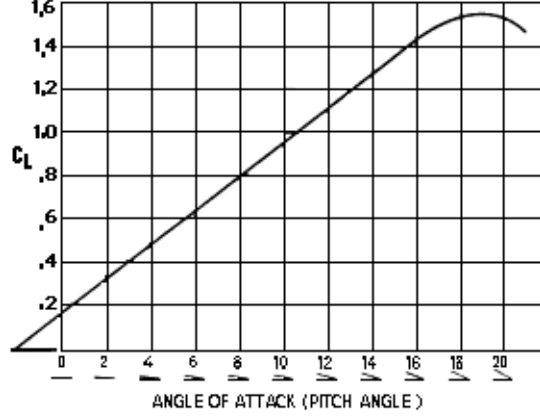


Figure 3.7: The propeller angle of attack changes because of linear speed

Assigning 3.22 into 3.30 we get:

$$\Delta L = \int \frac{1}{2} A_d (\dot{X}_N^2 + (\Omega \cdot r)^2) c(r) M_{cl} \arctan\left(\frac{\dot{X}_N}{\Omega \cdot r}\right) dr \quad (3.23)$$

In order to simplify this equation and respect the nonlinearity of the relations between the angle of attack and the lift coefficient, we will make the following approximations: $c(r)$ is constant, and the propeller doesn't produce thrust around its center (for small values of r the change in angle of attack as a result of \dot{X}_N is significant, and causes the blade to stall as seen in fig. 3.7 for $\alpha > 18[deg]$). Therefore $\dot{X}_N \ll \Omega \cdot r$ in the effective propeller area, which leads to $\arctan\left(\frac{\dot{X}_N}{\Omega \cdot r}\right) \rightarrow \frac{\dot{X}_N}{\Omega \cdot r}$ and $\dot{X}_N^2 + (\Omega \cdot r)^2 \rightarrow (\Omega \cdot r)^2$

$$\Delta L = \frac{1}{2} c M_{cl} A_d \dot{X}_N \Omega \int_{R_{min} \approx 0}^{R_b} r dr = \frac{1}{4} c M_{cl} A_d \dot{X}_N \Omega R_b^2 \quad (3.24)$$

Here R_b is the length of the propeller blade and R_{min} is the length from the rotation axis of the propeller to the point in which the propeller starts producing lift . In the example given here:

$$\begin{aligned} \dot{X}_{N1} &= \dot{x} \sin(\beta - \theta) \\ \dot{X}_{N3} &= -\dot{x} \sin(\beta + \theta) \end{aligned} \quad (3.25)$$

Here X_{N1} is X_N for propeller 1 and X_{N3} is X_N for propeller 3. In Order to calculate ΔL in the more general case, we first need to define the normal to each rotor disk - in the body system:

$$p1 = \begin{bmatrix} -\sin \beta \\ 0 \\ \cos \beta \end{bmatrix} \quad p2 = \begin{bmatrix} 0 \\ -\sin \beta \\ \cos \beta \end{bmatrix} \quad p3 = \begin{bmatrix} \sin \beta \\ 0 \\ \cos \beta \end{bmatrix} \quad p4 = \begin{bmatrix} 0 \\ \sin \beta \\ \cos \beta \end{bmatrix}$$

(3.26)

And in the fixed system:

$$\begin{aligned}
 P1 &= R \cdot p1 \\
 P2 &= R \cdot p2 \\
 P3 &= R \cdot p3 \\
 P4 &= R \cdot p4
 \end{aligned} \tag{3.27}$$

\dot{X}_{Ni} will be the dot product of P_i and the linear speed of the quadrotor:

$$\dot{X}_{Ni} = V P_i = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \cdot P_i \tag{3.28}$$

Using eq.3.28 in eq. 3.24 we get:

$$\Delta L_i = \frac{1}{4} c M_{cl} A_d (V \cdot P_i) \Omega R_b^2 \tag{3.29}$$

This change in lift creates another element of τ :

$$\begin{cases} \tau_{xl} = l(\Delta L3 - \Delta L1) \\ \tau_{yl} = l(\Delta L4 - \Delta L2) \end{cases} \tag{3.30}$$

And the total nonconservative torque:

$$\begin{cases} \tau_x = \tau_{xa} + \tau_{xd} + \tau_{xg} + \tau_{xl}(V) \\ \tau_y = \tau_{ya} + \tau_{yd} + \tau_{yg} + \tau_{yl}(V) \\ \tau_z = \tau_{za} + \tau_{zd} + \tau_{zg} \end{cases} \quad (3.31)$$

As before, the dynamic equations that describe attitude are presented in eq. 3.18, however, the nonconservative torque has a component which is a function of linear speed. When linear motion V has components in the horizontal plane XY , a torque will act on the hovercraft, rotating it in such a way that will cause it to accelerate in the opposite direction. This action is essentially a negative feedback which will help stabilize the hovercraft in the XY plane.

3.4. Position Dynamics of the QTRH

In the previous sections we discussed attitude modeling, but not the position dynamics. As is eq.3.5, we will assume that only gravity and actuator forces are acting on the robot. This results in a very similar model:

$$\begin{bmatrix} \ddot{X} \\ \ddot{Y} \\ \ddot{Z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} + R \begin{bmatrix} 0 \\ 0 \\ \frac{c\beta b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2)}{m} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \frac{c\beta u_1}{m} \end{bmatrix} \quad (3.32)$$

The tilted motors do not use all of their energy to produce lift or acceleration, thus reducing the robots efficiency, and maximum acceleration ability.

CHAPTER 4

Attitude Estimation

Attitude estimation is the heart of the system. During this research we discovered that the flight characteristics are extremely sensitive to changes in the quality of attitude estimation, much more than to control gains. In this chapter we will survey some algorithms that convert sensor data into attitude estimation. Understanding these algorithms is very important in order to assess the properties of the estimation.

4.1. Measurements and Sensors

An Inertial Measurement Unit (IMU) is the unit that estimates the attitude of the hovercraft. It usually includes 3 axes gyroscopes for angular speed measurements and three axes accelerometers used to measure the direction of the earth's gravity. Other common sensors include a 3 axis magnetometer for measuring the earth's magnetic field and a GPS unit for measuring location and speed. In this chapter we will discuss the algorithms used to calculate the orientation of the hovercraft relative to the earth.

Accelerometers - MEMS (micro electro mechanical sensors) accelerometers measure the difference between gravity and acceleration. If a three axis accelerometer is attached to a hovercraft, so that the

accelerometer axes are aligned with the hovercraft axes, the accelerometer measurement will be:

$$acc = R^T \left(\begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} - \begin{bmatrix} \ddot{X} \\ \ddot{Y} \\ \ddot{Z} \end{bmatrix} + \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} \right) \quad (4.1)$$

Where n denotes sensor noise. Assuming $\ddot{X} = 0$ and $n = 0$ we get:

$$\begin{aligned} acc_x &= r_{31}g \\ acc_y &= r_{32}g \\ acc_z &= r_{33}g \end{aligned} \quad (4.2)$$

From 4.2 it is possible to calculate roll and pitch angles:

$$\begin{aligned} \theta &= asin\left(\frac{-acc_x}{g}\right) \\ \phi &= atan2(acc_y, acc_z) \end{aligned} \quad (4.3)$$

However, using accelerometer alone for pitch and roll calculations is not possible in the case of most aircrafts, because the assumptions are not correct. First, $n \neq 0$. In fig 4.1 we can see that the vibrations caused by the propulsion system can severely corrupt the measurement. It is, however, possible to assume that the noise has a zero mean, or even measure it and correct the measurements accordingly. Secondly, although it is possible to assume that on average the accelerations are equal to zero, especially during hovering, $\ddot{X}(t) \neq 0$.

In conclusion, the measurement of low frequencies provided by the accelerometer is relatively reliable, in some cases, mainly when flying at

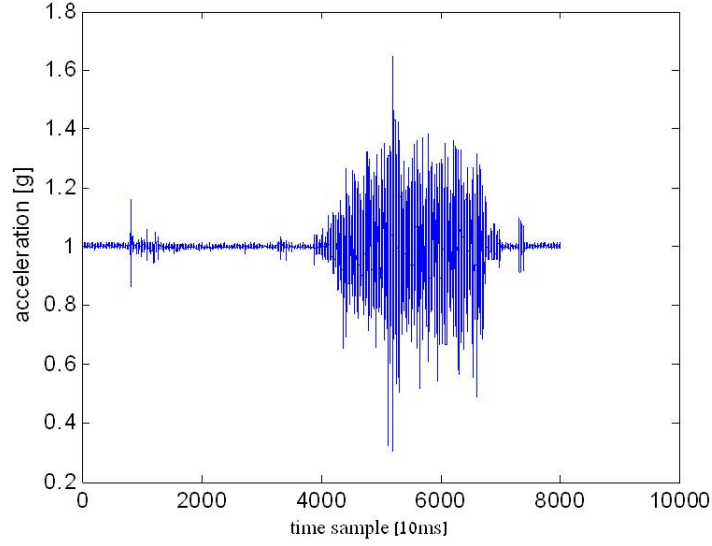


Figure 4.1: Accelerometer data along the z axis. In the first 4000 samples electric motors are at rest, and the reading is relatively accurate. At 4000 samples the motors start spinning gradually and the vibrations severely corrupt the data from the sensor.

a constant speed, and is very unreliable when the aircraft is in a state of constant acceleration. The high frequencies are usually completely unusable.

Magnetometers and yaw estimation- The magnetometer can be used in a similar manner as the accelerometers. The reading from the magnetometers can be modeled as:

$$mag = R^T(m + n) \quad (4.4)$$

Here m denotes the earth's magnetic field with magnitude $|m|$. If we choose A so that X is aligned with m , we get:

$$mag = R^T \begin{bmatrix} 0 \\ |m| \\ 0 \end{bmatrix} + n \quad (4.5)$$

Assuming that $n = 0$ we get:

$$\begin{aligned} mag_x &= r_{21}|m| \\ mag_y &= r_{22}g|m| \\ mag_z &= r_{23}g|m| \end{aligned} \quad (4.6)$$

From eq. 4.6 and eq.4.2 it is possible to calculate the yaw angle:

$$\phi = atan2\left(\frac{mag_x}{|m|}, \frac{acc_y}{|g|}\right) \quad (4.7)$$

Gyroscopes in state estimation - A rotating vector e with angular velocity w obeys the kinematics:

$$\dot{e} = w \times e \quad (4.8)$$

Applying 4.8 on the principal axes of system A , as seen in system B , we get:

$$\begin{aligned}
\dot{x}_{ba} &= w_{xba} \times x_{ba} \\
\dot{y}_{ba} &= w_{yba} \times y_{ba} \\
\dot{z}_{ba} &= w_{zba} \times z_{ba}
\end{aligned} \tag{4.9}$$

Here x_{ba}, y_{ba}, z_{ba} are A 's principle axes, as seen in B , and $w_{xba}, w_{yba}, w_{zba}$ are the measured angular speeds of system A compared to system B along the principle axes of B . Since the MEMS gyros that are attached to the body measure the body velocity in the body frame of reference, eq. 4.10 can be rewritten as:

$$\begin{aligned}
\dot{x}_{ba} &= x_{ba} \times w_{xab} \\
\dot{y}_{ba} &= y_{ba} \times w_{yab} \\
\dot{z}_{ba} &= z_{ba} \times w_{zab}
\end{aligned} \tag{4.10}$$

Here $w_{xba}, w_{yba}, w_{zba}$ are the measured body velocitys, as measured in the body system B . Since x_{ba}, y_{ba}, z_{ba} constitute the rows of the rotation matrix R , eq. 4.10 gets the form:

$$\dot{R} = R\hat{\omega} \tag{4.11}$$

Where $\hat{\omega}$ denotes the skew-symmetric (or anti-symmetric) matrix so that $\hat{\omega}a = \omega \times a$. By integrating eq.4.11, it is possible to track the rotation matrix, thus calculating pitch, roll and yaw angles using eq.2.9.

This method is not practical since gyro measurements are not free of noise, and bias:

$$\omega_g = \omega + b_i(t) + n_g \quad (4.12)$$

$b_i(t)$ is a slow changing bias, and n_g is measurement noise. The bias integration and other errors, such as discretization error and integration errors, cause drift, thus rendering the low frequencies of the estimation unreliable. However, the high frequencies of this estimation method stays sufficiently reliable.

Infrared thermopiles - An infrared thermopile is a sensor that can measure thermal energy[18]. Since the earth is warm (approx. $14^\circ C$) and the sky is cold (approx. $-40^\circ C$), they can be used to measure orientation by measuring the various temperatures in different directions. For example, if one sensor measures the temperature along the body's x_b axis positive direction, and another sensor measures the temperature in the negative direction the difference between the results will indicate the aircraft pitch angle. The sensors are very small and lightweight, and are very suitable for use in MAV's. The main drawback is that the measurement is highly dependant on environmental conditions - that is the exact warmth of the ground and skies under different weather conditions.

In this section we will propose a new method of measuring attitude using infrared thermopiles. For this purpose, we propose a devise that includes two thermopiles heading in opposite directions, as in the example above. The outputs of the sensors are differentiated and amplified

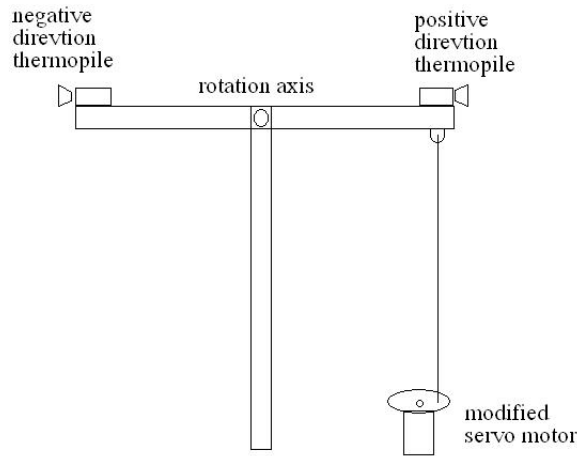


Figure 4.2: Thermopile based direct orientation measurement mechanism. The modified servo regulates the thermopiles orientation automatically.

using a differential amplifier [15]. Since the only position where the thermopiles measurement difference is constant under any ground to sky temperature gradient is when the thermopiles are horizontal - that is both thermopiles measure the same temperature, we suggest that the mechanism will automatically keep the thermopiles horizontal. To do so, the thermopiles will be mounted on an axis, where its position is controlled by a simple servo motor (fig.4.2).

Figure 4.3 shows a block diagram for a standard servo motor. A control signal containing the desired position of the servo arm is the input of the system. This input is pulse width modulated. The actual position of the motor is measured using a potentiometer. This measurement is then converted into pulse width modulation to create the error signal using a pulse width comparator. A small modification to the servo motor that controls the position of the thermopiles

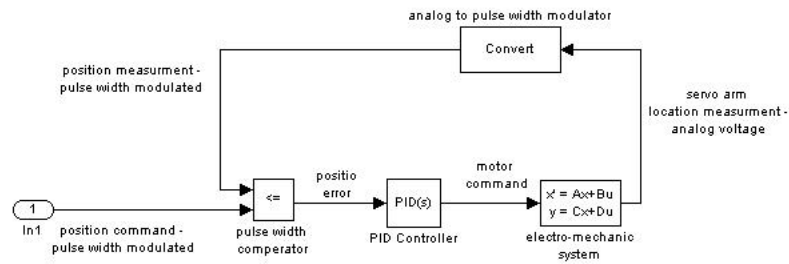


Figure 4.3: Standard servo motor block diagram

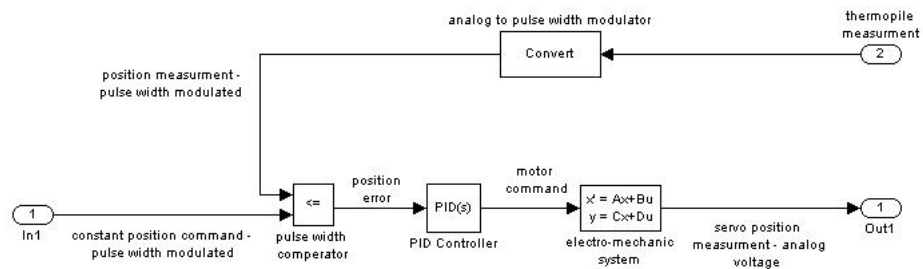


Figure 4.4: The modified servo motor block diagram

is presented in fig. 4.4. In this structure, the position feedback of the servo arm is replaced with the thermopile measurement (the output of the differential amplifier). Instead of controlling the servo position, the servo controller regulates the thermopile position. The servo arm position measurement is the output of the system and indicates the angle of the thermopiles relative to the body. When assuming that the thermopiles are horizontal, the output is a direct measurement of the body's orientation relative to the horizon. This mechanism can substitute the accelerometers direct pitch and roll measurement, with the advantage of not being affected by vibrations and accelerations.

4.2. Complementary filter with no bias estimation

In this simple approach we suggest calculating the relevant variables of R for roll pitch and yaw calculations using gyros and accelerometers for pitch and roll and magnetometers for yaw. We then use complementary filter on each of the variables separately to fuse the data from the gyros and accelerometers so that high frequency information is originated by the gyros and low frequency information is originated by the accelerometers. The filtered results are then used for euler angles calculation. The relevant variables are r_{31} for pitch, r_{32}, r_{33} for roll, and r_{21} for yaw. We use the following form of digital complementary filter:

$$\begin{aligned}\tilde{r}_{i,j} &= H_{hpf}r_{gi,j} + H_{lpf}r_{ai,j} = \tilde{r}_{gi,j} + \tilde{r}_{ai,j} \\ H_{hpf} &= \frac{\alpha - \alpha Z^{-1}}{1 - \alpha Z^{-1}} \\ H_{lpf} &= \frac{1 - \alpha}{1 - \alpha Z^{-1}}\end{aligned}\tag{4.13}$$

The high pass filter of the complementary filter is implemented using the following discrete iterative equation:

$$\tilde{r}_{gi,j}(k) = \alpha \tilde{r}_{gi,j}(k) + \alpha(r_{gi,j}(k) - r_{gi,j}(k-1))\tag{4.14}$$

Using the discrete form of eq 4.11:

$$R(k) = R(k-1)(I_3 + \hat{\omega}(k))dt\tag{4.15}$$

And the estimation based on gyro sensors:

$$\begin{aligned}\tilde{R}(k) &= \tilde{R}(k-1)(I_3 + \hat{\omega}_g(k))dt \\ \tilde{R}(k) - \tilde{R}(k-1) &= \tilde{R}(k-1)(\hat{\omega}_g(k))dt\end{aligned}\tag{4.16}$$

We get:

$$\begin{aligned}\tilde{r}_{g31}(k) &= \alpha\tilde{r}_{g31}(k-1) + \alpha(\tilde{r}_{g32}(k-1)\omega_{gz}(k) - \tilde{r}_{g33}(k-1)\omega_{gy}(k))dt \\ \tilde{r}_{g32}(k) &= \alpha\tilde{r}_{g32}(k-1) + \alpha(-\tilde{r}_{g31}(k-1)\omega_{gz}(k) + \tilde{r}_{g33}(k-1)\omega_{gx}(k))dt \\ \tilde{r}_{g33}(k) &= \alpha\tilde{r}_{g33}(k-1) + \alpha(\tilde{r}_{g31}(k-1)\omega_{gy}(k) - \tilde{r}_{g32}(k-1)\omega_{gx}(k))dt \\ \tilde{r}_{g21}(k) &= \alpha\tilde{r}_{g21}(k-1) + \alpha(\tilde{r}_{g22}(k-1)\omega_{gz}(k) - \tilde{r}_{g23}(k-1)\omega_{gy}(k))dt\end{aligned}\tag{4.17}$$

The low pass filter of the complementary filter is implemented using the following discrete iterative equation:

$$\tilde{r}_{ai,j}(k) = \alpha\tilde{r}_{ai,j}(k-1) + (1-\alpha)r_{ai,j}(k)\tag{4.18}$$

and specifically for each of the variables:

$$\begin{aligned}\tilde{r}_{a31}(k) &= \alpha\tilde{r}_{a31}(k-1) + (1-\alpha)acc_x(k) \\ \tilde{r}_{a32}(k) &= \alpha\tilde{r}_{a32}(k-1) + (1-\alpha)acc_y(k) \\ \tilde{r}_{a33}(k) &= \alpha\tilde{r}_{a33}(k-1) + (1-\alpha)acc_z(k) \\ \tilde{r}_{a21}(k) &= \alpha\tilde{r}_{a21}(k-1) + (1-\alpha)mag_x(k)\end{aligned}\tag{4.19}$$

The parameter α controls the cut-off frequency and is selected according to bias behaviour, linear accelerations and noise, usually experimentally. The main advantage of this method is its simplicity,

which also allows for short iteration times when implemented on low power processors. Its main disadvantage is that this method does not estimate and correct gyro bias. This makes gyro drift correction less efficient, and for good bias correction one needs to rely heavily on the accelerometers, thus increasing the undesired effects of externally caused accelerations and vibrations. Another disadvantage is that the angular velocity measurement needed by the controller for stabilization is less accurate due to the bias.

4.3. Kalman Filter Based Approach

The one dimensional case - Before discussing the full orientation estimation, we will discuss the one dimensional case in which there is only one axis of rotation. Specifically, we will consider the case in which this axis is perpendicular to earth's gravity, as in the case of the inverted pendulum. The sensors used to estimate the orientation are a single axis accelerometer and a single axis gyroscope. The approach is to replace the dynamic equation normally used to predict the state of the system with the kinematic equation which integrates data from the gyro for estimating the angle. However, since the gyro data, 4.12 has a bias component which cannot be treated as white gaussian noise, we will treat it as another state variable, the behavior of which we have no information about, except that it is slow changing. The prediction equation becomes:

$$\begin{bmatrix} \theta_p \\ b_p \end{bmatrix} (k+1) = \begin{bmatrix} 1 & -dt \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{\theta} \\ b \end{bmatrix} (k) + \begin{bmatrix} dt \\ 0 \end{bmatrix} \omega_g(k+1) \quad (4.20)$$

Here θ_p and b_p are the predicted angle and bias. For the measurement step, a direct measurement of the gravity projection on the accelerometer sensitivity axis is taken. In the case of a pendulum, the sensor is placed so that its sensitivity axis is perpendicular to the earth's gravity and to the rotation axis when the pendulum is at rest. In this configuration, the measurement equation of the Kalman filter is:

$$\theta_m = a \sin(-acc) \quad (4.21)$$

Here θ_m is the measured angle. The covariance matrixes required for the Kalman filtering algorithm can be experimentally deduced, while assuming that $E(n_\theta n_b) = 0$. The value of $E(n_b^2)$ determines the rate of bias change, where a low value will cause the filter to reject fast changes in bias.

The three dimensional case - In the three dimensional case, the prediction equation should be based on eq. 4.15, where the state variables are $r_{i,j}$. However, since gyro bias cannot be treated as white gaussian noise, we have to consider it as a state variable. The kinematic equation used to produce a prediction:

$$\tilde{R}(k) = \tilde{R}(k-1)(I_3 + (\hat{\omega}_g(k) - \hat{b}(k-1))dt) \quad (4.22)$$

One can see that this is not a linear equation, since it contains $\tilde{R}(k-1)\hat{b}(k-1)$ and, therefore, sensor fusion using Kalman filtering is not possible in this manner.

Extended Kalman Filtering - EKF can be used to overcome the nonlinearity of 4.22. In the same manner as in the one dimensional case, the input for the kinematic equation, which replaces the dynamic equation, is the data from the gyros, while the data from the accelerometers and magnetometers is considered as the measurement for the algorithm. The state vector is chosen as:

$$x = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 \end{bmatrix}^T = \begin{bmatrix} r_{2,1} & r_{3,1} & r_{3,2} & r_{3,3} & b_x & b_y & b_z \end{bmatrix}^T \quad (4.23)$$

The prediction step (based on 4.22):

$$x_p(k+1) = \begin{bmatrix} x_1(k) + r_{22}(k)(\omega_{gz}(k+1) - x_7(k))dt - r_{23}(k)(\omega_{gy}(k+1) - x_6(k))dt \\ x_2(k) + x_3(k)(\omega_{gz}(k+1) - x_7(k))dt - x_4(k)(\omega_{gy}(k+1) - x_6(k))dt \\ x_3(k) - x_2(k)(\omega_{gz}(k+1) - x_7(k))dt + x_4(k)(\omega_{gx}(k+1) - x_5(k))dt \\ x_4(k) + x_2(k)(\omega_{gy}(k+1) - x_6(k))dt - x_3(k)(\omega_{gx}(k+1) - x_5(k))dt \\ x_5(k) \\ x_6(k) \\ x_7(k) \end{bmatrix} \quad (4.24)$$

Where the state transition matrix is:

$$F = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & r_{23}dt & -r_{22}dt \\ 0 & 1 & f_{23} & f_{24} & 0 & x_4(k)dt & -x_3dt \\ 0 & f_{32} & 1 & f_{34} & x_5(k)dt & 0 & x_2dt \\ 0 & f_{42} & f_{43} & 1 & x_3dt & -x_2dt & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.25)$$

$$f_{23} = (\omega_{gz}(k+1) - x_7(k))dt$$

$$f_{24} = -(\omega_{gy}(k+1) - x_6(k))dt$$

$$f_{32} = -(\omega_{gz}(k+1) - x_7(k))dt$$

$$f_{34} = (\omega_{gx}(k+1) - x_5(k))dt$$

$$f_{42} = (\omega_{gy}(k+1) - x_6(k))dt$$

$$f_{43} = -(\omega_{gx}(k+1) - x_5(k))dt$$

And since the magnetometers and accelerometers provide a direct measurement of the state variables, the observation matrix is:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \quad (4.26)$$

Using equations 4.24 and the definitions of H and F , it is simple to implement the EKF algorithm and obtain a near optimal bias and orientation estimation (under the assumption of constant bias).

4.4. State Estimation Coupled With System Dynamics

In most works we have encountered, state estimation is based solely on kinematic models, and does not take into account the dynamic behaviour of the aircraft [35, 34, 36]. In this section, we will analyze the accelerometer roll in state estimation as used in most algorithms, while taking into account the effects of the aircraft's dynamic behaviour. First, we will use the model presented in 3.7, and later we will consider the specific case of a banked level turn (coordinated turn), which is typical in fixed wing aircrafts or in a forward flight of a quadrotor.

Accelerometer's role in Quadrotor state estimation - If one takes into account only the internal forces acting on the robot (forces caused by the actuators) and neglects the forces created by drag, as is valid for low speeds (like in the case of hovering), by substituting 3.5 into 4.1 we get:

$$acc = R^T \left(\begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} - \left(\begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} + R \begin{bmatrix} 0 \\ 0 \\ \frac{u_1}{m} \end{bmatrix} \right) + \begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} \right) \quad (4.27)$$

Assuming no noise, we get:

$$acc = \begin{bmatrix} 0 \\ 0 \\ \frac{u_1}{m} \end{bmatrix} \quad (4.28)$$

Since the accelerometer measurement is not a function of the rotation matrix, no orientation data can be acquired under the assumption of no drag. The acceleration resulting from pitch and roll angles is mixed in the sensor reading in a way that completely masks the gravitation measurement. In the case where drag is considerable, it will reduce acceleration to zero for any given angle, thus making the accelerometers reading for gravity accurate. It is worth noting that according to the drag equation:

$$F_D = \frac{1}{2} A_d v^2 d A \quad (4.29)$$

Here F_D is the drag force, A_d is air density, d is the drag coefficient, A is the reference area, and v is air speed, the drag is proportional to the squared air speed, thus it can only be neglected at very low speeds, and only for short periods of time. That is the case in a hover, since the direction of acceleration keeps changing at opposite directions around a fixed reference point, while the linear velocities remain low.

Accelerometer's roll in state estimation while flying in a banked level turn - Long periods of constant acceleration while flying along a straight line is not possible due to drag. However it is possible and very common in the case of a coordinated turn. In this case, we assume constant aircraft speed, which means that the drag is equal to the thrust, and thus $acc_x = 0$. Figure 4.5 shows the forces acting on the aircraft along the y_b, z_b plain. The radius of the turn can be calculated by:

$$R = \frac{mv^2}{f} \quad (4.30)$$

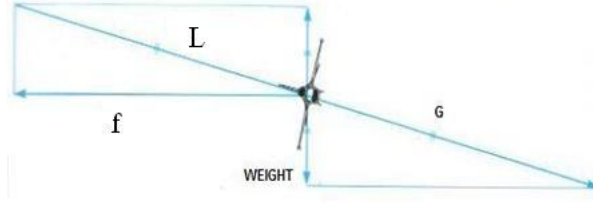


Figure 4.5: Forces in the z_b, y_b plain in a coordinated turn of an aircraft with mass m

and the force f is a function of the roll angle:

$$f = L \sin(\phi) = mg \tan \phi \quad (4.31)$$

The aircraft is in a state of constant acceleration in the direction of f , and the acceleration magnitude is $\frac{f}{m}$. To simplify the calculations, we will consider the two dimensional problem in the y_b, z_b plain. As mentioned in eq.4.1, the accelerometers measure the difference between gravity and acceleration:

$$\begin{aligned} \begin{bmatrix} acc_y \\ acc_z \end{bmatrix} &= R^T \left(\begin{bmatrix} 0 \\ g \end{bmatrix} - \begin{bmatrix} g \tan \phi \\ 0 \end{bmatrix} \right) = \\ &= \begin{bmatrix} c\phi & s\phi \\ -s\phi & c\phi \end{bmatrix} \begin{bmatrix} -g \tan \phi \\ g \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{g}{\cos \phi} \end{bmatrix} \end{aligned} \quad (4.32)$$

Since there are no accelerations along the x_b axis, and from assuming

$$\theta = 0 \text{ we get } \begin{bmatrix} acc_x \\ acc_y \\ acc_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \frac{g}{\cos \phi} \end{bmatrix}. \text{ This means that during a coordinated}$$

turn the magnitude of the bank (ϕ) can be calculated as:

$$|\phi| = \cos^{-1} \left(\frac{g}{acc_z} \right) \quad (4.33)$$

but the sign of ϕ cannot be determined. Note that this is a very different result than the one presented in eq. 4.3, where the dynamic behavior was not taken into account.

4.5. Dynamic Damping

One of the problems encountered while working with accelerometers is that they are very sensitive to vibrations. Vibrations can be caused by unbalanced motors and propellers or by an internal combustion engine mounted on the robot. Figure 4.1 clearly shows the effects of these vibrations on the accelerometers. In order to reduce these vibrations we have designed an instrument which uses the dynamic damping presented in chapter 2.5 with the concept expanded to three dimensions. The chassis (part 1) and absorbtion pad (part 8) are the equivalent of the mass m_1 and the spring k_1 in fig 2.2. The springs (parts 2,4,6) are the equivalent of the spring k_2 . The small masses (parts 3,5,7) are the equivalent of mass m_2 . The device is attached to the robot via an absorbtion pad (part 8), and the sensor board containing the accelerometers (part 9) is mounted on top of the chassis. The resonance frequency of each of the three mass-spring pairs (parts 2 and 3, 4 and 5, 6 and 7) can be adjusted by changing the mass of each of the weights, or by changing its position on the spring. Changing the weight position changes the spring length, and as a result changes the spring constant. Each of the spring-weight pairs dissipates energy along a different axis, for example the pair of parts 6 and 7 dissipates vibrations along the body axis y_b

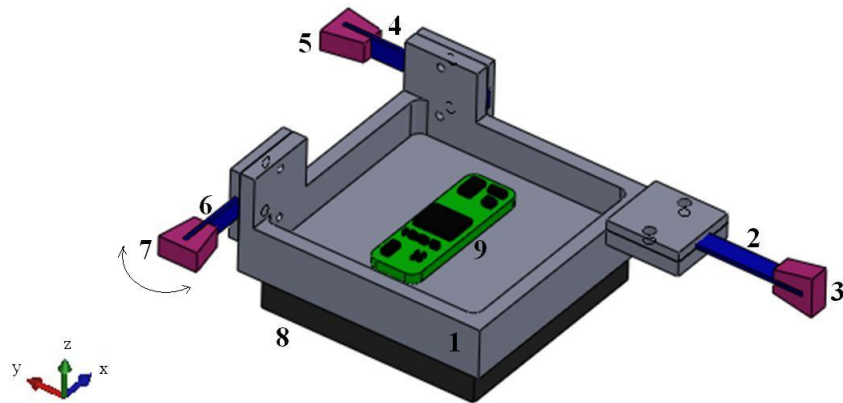


Figure 4.6: Dynamic Damping mechanism for an Inertial Measurement Unit. 1 - Device chassis. 2,4,6 - Springs. 3,5,7 - Weights. 8 - Shock absorbent pad. 9 - sensor board

CHAPTER 5

Control

The quadrotor is an unstable under actuated system. It contains the attitude sub-system which is independent of the position of the system. In this chapter we will discuss the methods of stabilizing both the position and attitude of the quadrotor using linear and nonlinear methodologies.

5.1. Attitude control analysis based on the linear model

Proportional attitude controller - As described in eq. 3.11, the quadrotor attitude dynamic model can be approximated as three independent, almost identical, linear subsystems.

$$\begin{aligned}\Phi &= \frac{l}{I_x} \frac{1}{(\tau s + 1)} \frac{1}{s^2} U_{2ref} \\ \Theta &= \frac{l}{I_y} \frac{1}{(\tau s + 1)} \frac{1}{s^2} U_{3ref} \\ \Psi &= \frac{1}{I_z} \frac{1}{(\tau s + 1)} \frac{1}{s^2} U_{4ref}\end{aligned}$$

Where Φ, Θ, Ψ are the Laplace transforms of the roll pitch and yaw angles (ϕ, θ, ψ) respectively, and $U_{iref}, i = 2, 3, 4$ are the Laplace transforms of $u_{iref}, i = 2, 3, 4$ which are a function of the desired propeller velocities, as described in eq. 3.6. Root locus plot shows that when

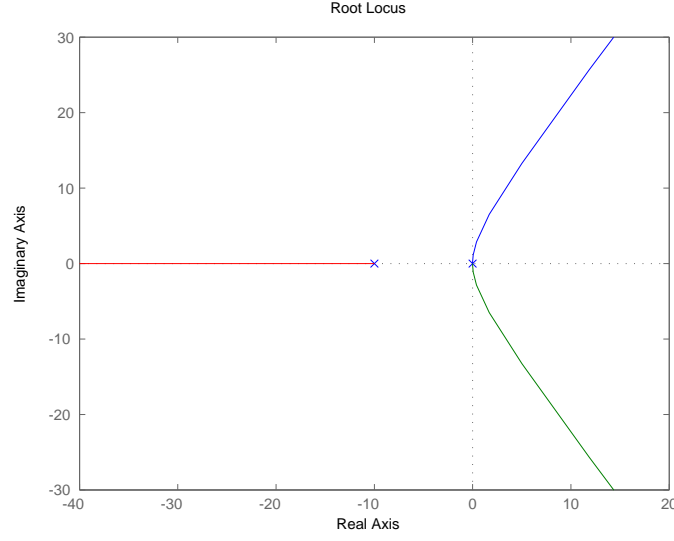


Figure 5.1: Root locus plot. Simple proportional feedback cannot stabilize the attitude

using output feedback (feedback proportional to the roll pitch and yaw angles only) on each of the subsystems, the attitude cannot be stabilized, as seen in figure 5.1.

Proportional plus Differential feedback - Because proportional feedback is not enough to stabilize the system, a differential feedback component is added, as described in figure 5.1

In this case U_{iref} get the form:

$$U_{iref} = Kp(X_{iref} - X_i) - s \cdot Kd \cdot X_i; i = 1, 2, 3 \quad (5.1)$$

Where X_i denotes roll pitch or yaw . For the rest of the chapter, we will continue the analysis regarding Φ only; however, the case is similar for the other two axis. Using eq. 3.11 we get:

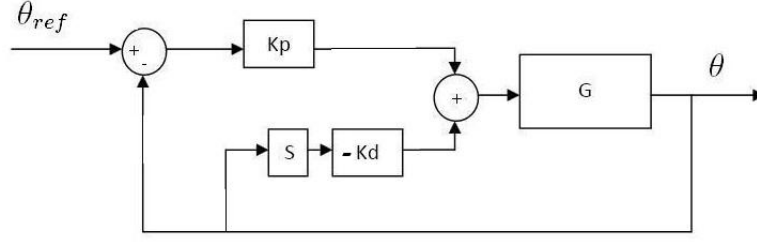


Figure 5.2: PD controller block diagram. G is one of the three sub-systems described by 3.11. K_p and K_d are the proportional and differential gains, respectively.

$$\Phi = \frac{l}{I_x} \frac{1}{(\tau s + 1)} \frac{1}{s^2} (K_p(\Phi_{ref} - \Phi) - s \cdot K_d \cdot \Phi) \quad (5.2)$$

And the transfer function:

$$\frac{\Phi}{\Phi_{ref}} = \frac{lK_p}{s^3 I_x \tau + s^2 I_x + s \cdot lK_d + lK_p} \quad (5.3)$$

In fig 5.3 we can see a zero pole map of the system for some values of K_p and K_d . A correct selection of K_p and K_d will stabilize the system. For relatively small values of K_p , we can see that the dominate poll is the pole located on the real axis, and when increasing K_p the dominate poles are the complex pole pair.

In fig. 5.4 we can see the dominant pole real value as a function of K_d and K_p . For fast settling time the best results will be when choosing the gains that correspond to the minimum points of this plot. A plot of K_p as a function of K_d that satisfies this requirement is seen in fig 5.5. The corresponding poles are seen in fig. 5.6. From these

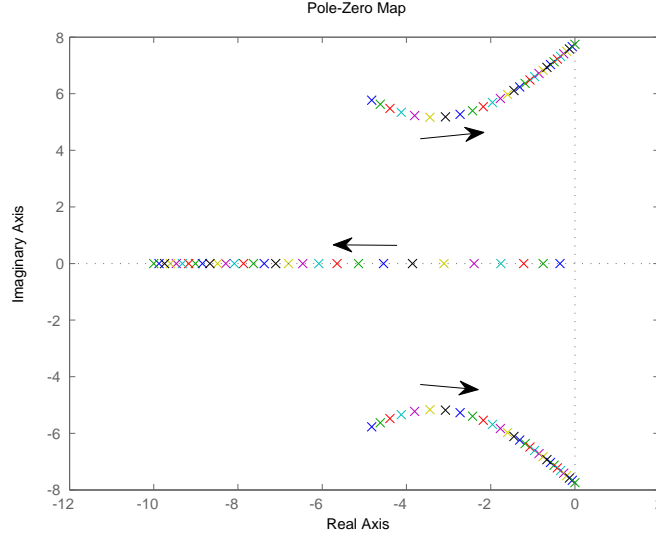


Figure 5.3: Zero-Pole map. $K_d = 0.3$ while K_p changes between 0.1 to 3

simulation results we can see that the best gains provide poles real value of approximately -3.5.

Acceleration feedback - Adding acceleration feedback yields the following control law:

$$U_i = K_p(X_{iref} - X_i) - (s^2 \cdot K_{dd} + s \cdot K_d)X_i; i = 1, 2, 3 \quad (5.4)$$

And as before we get:

$$U_2 = K_p(\Phi_{ref} - \Phi) - (s^2 \cdot K_{dd} + s \cdot K_d)\Phi; \quad (5.5)$$

$$\Phi = \frac{l}{I_x} \frac{1}{(\tau s + 1)} \frac{1}{s^2} (K_p(\Phi_{ref} - \Phi) - (s^2 \cdot K_{dd} + s \cdot K_d)\Phi) \quad (5.6)$$

$$\Phi = \frac{l K_p}{s^3 I_x \tau + s^2 (I_x + l K_{dd}) + s \cdot l K_d + l K_p} \quad (5.7)$$

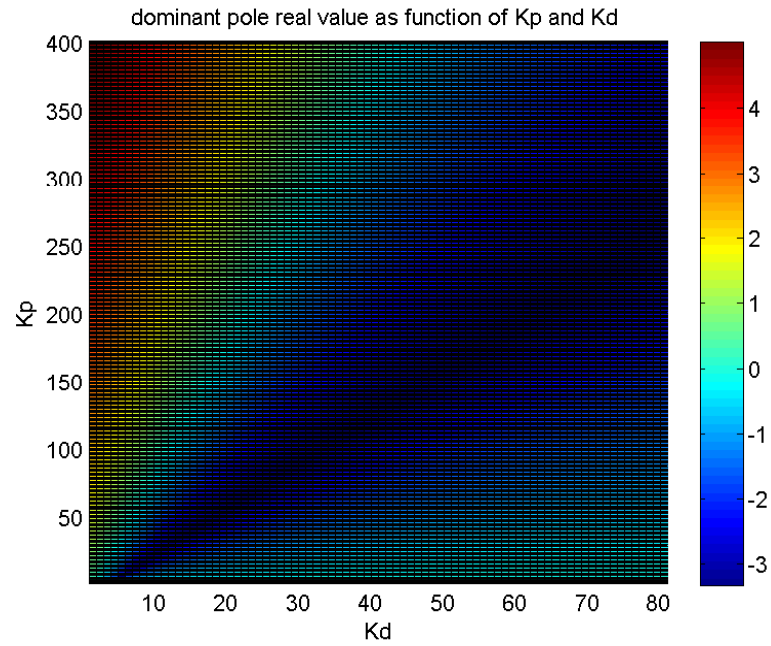


Figure 5.4: Dominant pole real value as a function of K_p and K_d as computed in simulation

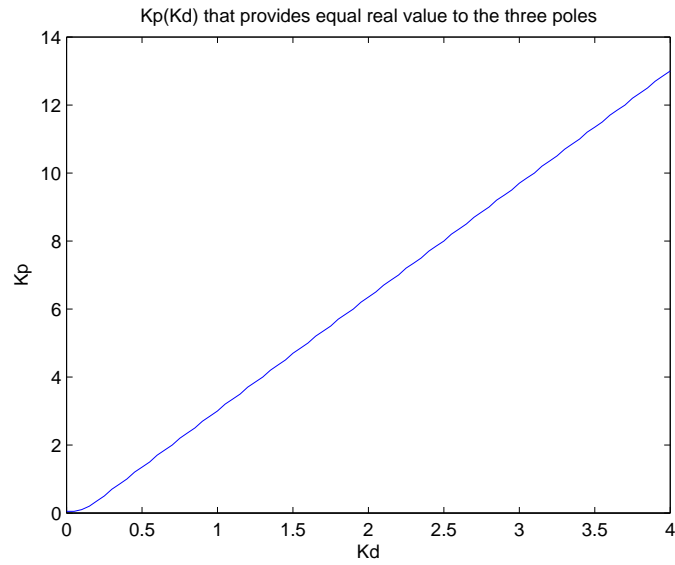


Figure 5.5: K_p as a function of K_d which gives closest real value for the three poles in simulation

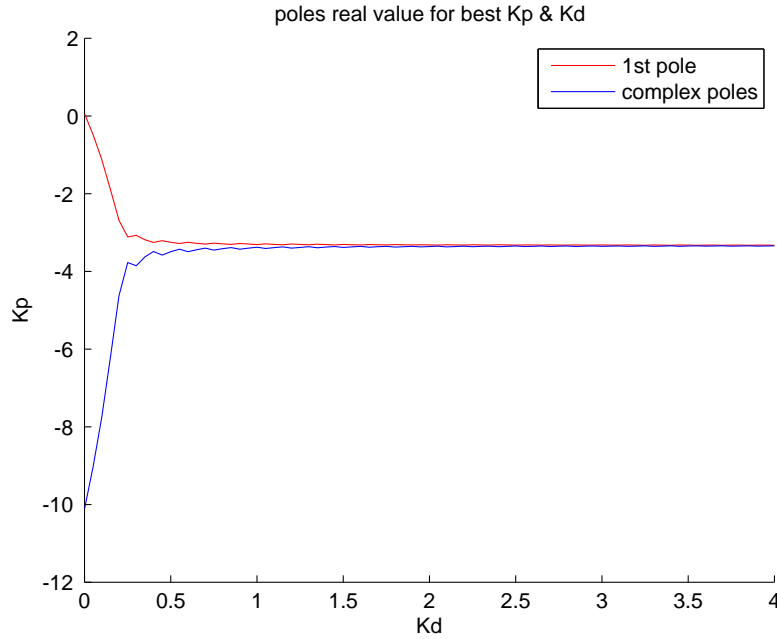


Figure 5.6: K_p as a function of K_d which gives closest real value for the three poles in simulation

Figure 5.7 is the corresponding figure to fig. 5.4 with the difference of adding acceleration feedback ($K_{dd} = 0.1$) to the controller. The changes in dominant pole real value as a function of K_d and K_p are very similar to the case without acceleration feedback.

Fig 5.8 shows K_p as a function of K_d which yields the closest real value for the three poles, and fig 5.9 shows the corresponding pole real values. It is clear from comparing fig 5.9 with fig.5.6 that for the same K_p and K_d the convergence time will be much better when adding acceleration feedback, since the real value of the dominant pole is much smaller. That is -10 compared to -3.5 .

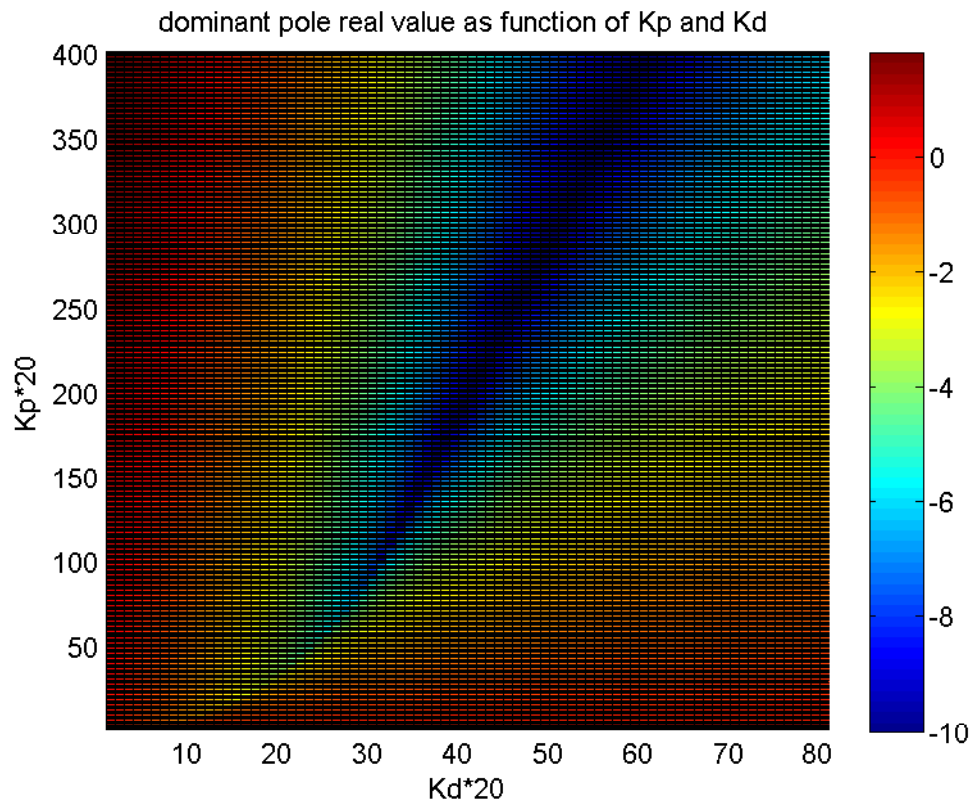


Figure 5.7: Dominant pole real value as a function of K_p and K_d as computed in simulation

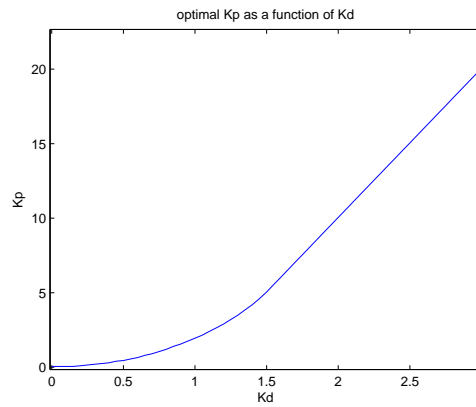


Figure 5.8: K_p as a function of K_d which gives closest real value for the three poles in simulation

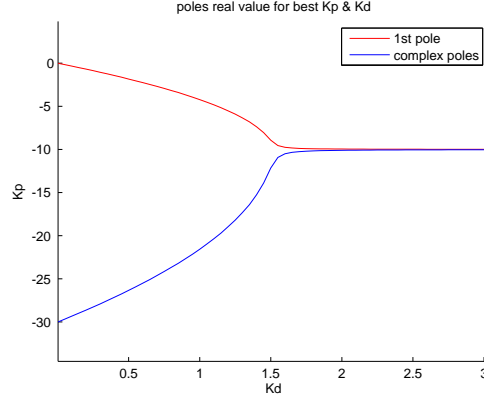


Figure 5.9: K_p as a function of K_d which gives closest real value for the three poles in simulation

5.2. Attitude control analysis based on the non-linear model

In this section we will examine the stability of the non-linear model for attitude:

$$\begin{cases} I_x \ddot{\phi} = \dot{\theta} \dot{\psi} (I_y - I_z) + u_2 - J \dot{\theta} \Omega_r \\ I_y \ddot{\theta} = \dot{\phi} \dot{\psi} (I_z - I_x) + u_3 + J \dot{\phi} \Omega_r \\ I_z \ddot{\psi} = \dot{\phi} \dot{\psi} (I_x - I_y) + u_4 \end{cases} \quad (5.8)$$

under different control laws. We'll define $\zeta \triangleq \begin{bmatrix} \zeta_1 & \zeta_2 & \zeta_3 & \zeta_4 & \zeta_5 & \zeta_6 \end{bmatrix}^T \triangleq \begin{bmatrix} \phi & \dot{\phi} & \theta & \dot{\theta} & \psi & \dot{\psi} \end{bmatrix}^T$ the state vector. The model given by 5.8 can be

rewritten in state space as:

$$\begin{cases} \dot{\zeta}_1 = \zeta_2 \\ \dot{\zeta}_2 = \frac{1}{I_x}(\zeta_4\zeta_6(I_y - I_z) + u_2 - J\zeta_4\Omega_r) \\ \dot{\zeta}_3 = \zeta_4 \\ \dot{\zeta}_4 = \frac{1}{I_y}(\zeta_2\zeta_6(I_z - I_x) + u_3 + J\zeta_2\Omega_r) \\ \dot{\zeta}_5 = \zeta_6 \\ \dot{\zeta}_6 = \frac{1}{I_z}(\zeta_2\zeta_4(I_x - I_y) + u_4) \end{cases} \quad (5.9)$$

For stability analysis we'll choose a Lyapunov candidate function:

$$L = \frac{1}{2}(k_1\zeta_1^2 + k_2\zeta_2^2 + k_3\zeta_3^2 + k_4\zeta_4^2 + k_5\zeta_5^2 + k_6\zeta_6^2) \quad (5.10)$$

Where $k_i > 0$. In this case $L(0) = 0$ and $L(\zeta) > 0$ for any $\zeta \neq 0$. The second condition for asymptotic stability is that $\dot{L} < 0$ for any $\zeta \neq 0$:

$$\dot{L} = k_1\zeta_1\dot{\zeta}_1 + \dots + k_6\zeta_6\dot{\zeta}_6 \quad (5.11)$$

Assigning 5.9 to 5.11 we get:

$$\begin{aligned} \dot{L} = & k_2\zeta_2 \frac{1}{I_x} \left(\frac{k_1}{k_2} I_x \zeta_1 + \zeta_4 \zeta_6 (I_y - I_z) + u_2 - J\zeta_4 \Omega_r \right) + \\ & k_4\zeta_4 \frac{1}{I_y} \left(\frac{k_3}{k_4} I_y \zeta_3 + \zeta_2 \zeta_6 (I_z - I_x) + u_3 + J\zeta_2 \Omega_r \right) + \\ & k_6\zeta_6 \frac{1}{I_z} \left(\frac{k_5}{k_6} I_z \zeta_5 + \zeta_2 \zeta_4 (I_x - I_y) + u_4 \right) \end{aligned} \quad (5.12)$$

Nonlinear control law - A simple approach is to choose $u_{2,3,4}$ that eliminates the sign indefinite parts of the Lyapunov function, making

it negative semi-definite :

$$\begin{aligned}
 u_2 &= -\left(\frac{k_1}{k_2}I_x\zeta_1 + \zeta_4\zeta_6(I_y - I_z) - J\zeta_4\Omega_r\right) - \zeta_2 \\
 u_3 &= -\left(\frac{k_3}{k_4}I_y\zeta_3 + \zeta_2\zeta_6(I_z - I_x) + u_3 + J\zeta_2\Omega_r\right) - \zeta_4 \\
 u_4 &= -\left(\frac{k_5}{k_6} \zeta_5 + \zeta_2\zeta_4(I_x - I_y)\right) - \zeta_6
 \end{aligned} \tag{5.13}$$

Assigning 5.13 into 5.12 we get:

$$\dot{L} = -\frac{k_2}{I_x}\zeta_2^2 - \frac{k_4}{I_y}\zeta_4^2 - \frac{k_6}{I_z}\zeta_6^2 \tag{5.14}$$

Since \dot{L} is not a function of $\zeta_{1,3,5}$ it is only negative semi-definite. So in order to show asymptotic stability of the origin, we will use LaSalle's theorem. First we will assign 5.13 into 5.9 in order to get the autonomous system dynamic equation:

$$\begin{cases} \dot{\zeta}_1 = \zeta_2 \\ \dot{\zeta}_2 = -\frac{k_1}{k_2}\zeta_1 - \frac{\zeta_2}{I_x} \\ \dot{\zeta}_3 = \zeta_4 \\ \dot{\zeta}_4 = -\frac{k_3}{k_4}\zeta_3 - \frac{\zeta_4}{I_y} \\ \dot{\zeta}_5 = \zeta_6 \\ \dot{\zeta}_6 = -\frac{k_5}{k_6}\zeta_5 - \frac{\zeta_6}{I_z} \end{cases} \tag{5.15}$$

According to LaSalle's invariant set theory (section 2.6) in the case where R is the set of all points, where $\dot{V}(x) = 0$, that is $R = \{x \in \mathbb{R}^n : \dot{V}(x) = 0\}$ and M is the largest invariant set in R , and if M contains only the origin, then the origin is asymptotically stable. In our case \dot{L} is not a function of $\zeta_{1,3,5}$ and therefor $R = \{\zeta \in \mathbb{R}^6 : \dot{V}(\zeta) = 0\}$ that is $R = \{\zeta \in \mathbb{R}^6 : \zeta_2 = 0, \zeta_4 = 0, \zeta_6 = 0\}$. It is clear that if $\zeta_{2,4,6} \equiv 0$

then $\dot{\zeta}_{2,4,6} \equiv 0$. By assigning $\zeta_{2,4,6} \equiv 0$ and $\dot{\zeta}_{2,4,6} \equiv 0$ into eq. 5.15 we get $\zeta_{1,3,5} \equiv 0$ and $\dot{\zeta}_{1,3,5} \equiv 0$, thus $M = \{0, 0, 0, 0, 0, 0\}$ and the origin is asymptotically stable.

Feedback Linearization - Another approach is to choose $u_{2,3,4}$ such that eliminates the non linear parts of the system, making it linear:

$$\begin{aligned} u_2 &= -\zeta_4\zeta_6(I_y - I_z) + J\zeta_4\Omega_r - I_x(k_p\zeta_1 + k_d\zeta_2) \\ u_3 &= -\zeta_2\zeta_6(I_z - I_x) - J\zeta_2\Omega_r - I_y(k_p\zeta_3 + k_d\zeta_4) \\ u_4 &= -\zeta_2\zeta_4(I_x - I_y) - I_z(k_p\zeta_5 + k_d\zeta_6) \end{aligned} \quad (5.16)$$

Assigning 5.16 into 5.9 we get the following linear system:

$$\begin{cases} \dot{\zeta}_1 = \zeta_2 \\ \dot{\zeta}_2 = -(k_p\zeta_1 + k_d\zeta_2) \\ \dot{\zeta}_3 = \zeta_4 \\ \dot{\zeta}_4 = -(k_p\zeta_3 + k_d\zeta_4) \\ \dot{\zeta}_5 = \zeta_6 \\ \dot{\zeta}_6 = -(k_p\zeta_5 + k_d\zeta_6) \end{cases} \quad (5.17)$$

This system is composed of three independent subsystems of the form:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k_p & -k_d \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (5.18)$$

For stability, the eigenvalues of the matrix need to be with a negative real part. The eigenvalues are:

$$\lambda_{1,2} = -\frac{1}{2}k_d \pm \frac{1}{2}\sqrt{k_d^2 - 4k_p} \quad (5.19)$$

Correct selection of k_p and k_d will stabilize the system. For example $k_d > 0$ and $k_p = \frac{k_d^2}{4}$ will result in $\lambda_1 = \lambda_2 = -\frac{1}{2}k_d$.

PD controller - The most common controller used for attitude stabilization is a simple PD controller. In this section we will analyze the effect of this controller on the nonlinear attitude model. The PD control law is:

$$\begin{aligned} u_2 &= -K_{pr}(\phi - \phi_{ref}) - K_{dr}\dot{\phi} \\ u_3 &= -K_{pp}(\theta - \theta_{ref}) - K_{dp}\dot{\theta} \\ u_4 &= -K_{py}(\psi - \psi_{ref}) - K_{dy}\dot{\psi} \end{aligned} \quad (5.20)$$

In the case of a regulator we use a control law that stabilizes the system around $\zeta = 0$. The control law in state space becomes:

$$\begin{aligned} u_2 &= -K_{pr}\zeta_1 - K_{dr}\zeta_2 \\ u_3 &= -K_{pp}\zeta_3 - K_{dp}\zeta_4 \\ u_4 &= -K_{py}\zeta_5 - K_{dy}\zeta_6 \end{aligned} \quad (5.21)$$

Assigning 5.21 to eq. 5.9 we get:

$$\left\{ \begin{aligned} \dot{\zeta}_1 &= \zeta_2 \\ \dot{\zeta}_2 &= \frac{1}{I_x}(\zeta_4\zeta_6(I_y - I_z) - K_{pr}\zeta_1 - K_{dr}\zeta_2 - J\zeta_4\Omega_r) \\ \dot{\zeta}_3 &= \zeta_4 \\ \dot{\zeta}_4 &= \frac{1}{I_y}(\zeta_2\zeta_6(I_z - I_x) - K_{pp}\zeta_3 - K_{dp}\zeta_4 + J\zeta_2\Omega_r) \\ \dot{\zeta}_5 &= \zeta_6 \\ \dot{\zeta}_6 &= \frac{1}{I_z}(\zeta_2\zeta_4(I_x - I_y) - K_{py}\zeta_5 - K_{dy}\zeta_6) \end{aligned} \right. \quad (5.22)$$

We will show local stability at the origin using linearization. The linearized system is:

$$\dot{\zeta} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{-K_{pr}}{I_x} & \frac{-K_{dr}}{I_x} & 0 & \frac{-J\Omega_r}{I_x} - \zeta_6 & 0 & \zeta_4 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \zeta_6 + \frac{J\Omega_r}{I_y} & \frac{-K_{pp}}{I_y} & \frac{-K_{dp}}{I_y} & 0 & \zeta_2 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & \frac{-K_{py}}{I_z} & \frac{-K_{dy}}{I_z} \end{bmatrix} \zeta \quad (5.23)$$

And at the origin - $\zeta = 0$ we get:

$$\dot{\zeta} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{-K_{pr}}{I_x} & \frac{-K_{dr}}{I_x} & 0 & \frac{-J\Omega_r}{I_x} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{J\Omega_r}{I_y} & \frac{-K_{pp}}{I_y} & \frac{-K_{dp}}{I_y} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & \frac{-K_{py}}{I_z} & \frac{-K_{dy}}{I_z} \end{bmatrix} \zeta \quad (5.24)$$

In order to show local stability we need to show that the eigenvalues of the matrix are in the left half plane. The parametric solution for the eigenvalues is a very complicated expression and is not helpful in examining stability. Instead we will use estimated numeric values for I_x, I_y, I_z and J . Ω_r is a result of u_4 which is a linear combination of ζ_5 and ζ_6 . Since the linearization is around the origin, $\zeta_5 = \zeta_6 = 0$ we can assign $\Omega_r = 0$. In order to estimate moments of inertia we assumed that each motor weighs $0.06[Kg]$ and neglected the other mass elements. We estimate the distance of the motor from the center of mass to be $0.3[m]$. That results in:

$$I_x = 0.036 * 0.9[mKg]$$

$$I_y = 0.036[mKg]$$

$$I_z = 0.036 * 2 * 0.8[mKg]$$

We estimate the propeller inertia to be - $J = 0.1 * 0.005[mKg]$ according to $10[cm]$ blade length $0.001[Kg]$ mass, and the assumption of uniform mass distribution. The eigenvalues of this system when using unity PD feedback are:

$$\lambda_1 = -1.08$$

$$\lambda_2 = -12.8$$

$$\lambda_3 = -1.03 + 0.0005i$$

$$\lambda_4 = -1.03 - 0.0005i$$

$$\lambda_5 = -26.73 + 0.0005i$$

$$\lambda_6 = -26.73 - 0.0005i$$

We will also show stability of PD control law in simulation and in experiments in chapters [6](#) and [7](#).

5.3. Position Regulator

In this section we will present an approach for full control of a quadrotor robot to a hover state. We assume a regulation problem - the required hover position is the earth system A origin. We will use the linear attitude model from eq. [3.10](#) with the position dynamics

from 3.5 to get the following model:

$$\begin{cases} \ddot{\phi} = \frac{l}{I_x} u_2 \\ \ddot{\theta} = \frac{l}{I_y} u_3 \\ \ddot{\psi} = \frac{1}{I_z} u_4 \\ \ddot{Z} = -g + (c\phi c\theta) \frac{u_1}{m} \\ \ddot{X} = (c\phi s\theta c\psi + s\phi s\psi) \frac{u_1}{m} \\ \ddot{Y} = (c\phi s\theta s\psi - s\phi c\psi) \frac{u_1}{m} \end{cases} \quad (5.25)$$

The simplest approach for achieving a hover is the backstepping approach, in which a reference roll pitch and yaw angles $(\psi_{ref}, \phi_{ref}, \theta_{ref})$ are determined as a function of the position error (X and Y), and then u_2, u_3 and u_4 are determined as a function of the attitude error as described in previous sections. u_1 stabilizes the quadrotor along the Z axis. The position subsystem is over-actuated. There are three inputs and only two outputs. Therefor it is possible to set one of three reference angles to zero, and calculate the other two as a function the position error. For example, we can set ψ_{ref} to zero and then determine ϕ_{ref} as a function of position error along Y axis and θ_{ref} as a function of position error along the X axis. Another option is to set ϕ_{ref} to zero and then determine ψ_{ref} as a function of heading error and θ_{ref} as a function of distance from the origin of A . Here we will discuss the first option, and use a simple PD controller as follows:

$$\begin{cases} \theta_{ref} = -K_{pX}X - K_{dX}\dot{X} \\ \phi_{ref} = -K_{pY}Y - K_{dY}\dot{Y} \\ \psi_{ref} = 0 \end{cases} \quad (5.26)$$

Assigning the reference roll pitch and yaw from eq. 5.26 into eq.5.25 we get the following subsystem:

$$\begin{cases} \ddot{X} = (\cos(-K_{pY}Y - K_{dY}\dot{Y}) \sin(-K_{pX}X - K_{dX}\dot{X})) \frac{u_1}{m} \\ \ddot{Y} = -\sin(-K_{pY}Y - K_{dY}\dot{Y}) \frac{u_1}{m} \end{cases} \quad (5.27)$$

To simplify the analysis we will assume small ϕ_{ref} and θ_{ref} and we'll get:

$$\begin{cases} \ddot{X} = (-K_{pX}X - K_{dX}\dot{X}) \frac{u_1}{m} \\ \ddot{Y} = (K_{pY}Y + K_{dY}\dot{Y}) \frac{u_1}{m} \end{cases} \quad (5.28)$$

Stability can now be shown using Lyapunov criteria, but since the resulting system is linear if treating u_1 as constant it is easier to show stability using Laplace transform:

$$\begin{cases} X(s) = \frac{1}{s^2 + (-K_{dX}s - K_{pX}) \frac{u_1}{m}} \\ Y(s) = \frac{1}{s^2 + (K_{dY}s + K_{pY}) \frac{u_1}{m}} \end{cases} \quad (5.29)$$

This subsystem is stable, and poll location is easily determined by the coefficient values. A very important fact is that without differential feedback (K_{dY} and K_{dX}) the polls will be located on the imaginary axes in s domain, and the system will not be asymptotically stable. This is because the model does not include drag, and the differential feedback "creates" artificial friction to compensate. During hovering, due to

small speeds, the drag is very low, and this differential feedback is very important to damp the system. This concept stands behind the QTRH design. Its special structure creates this feedback mechanically, without controller intervention, thus eliminating the need for an accurate speed measurement.

QTRH Position Regulator - In this section we will consider a simplified one dimensional problem, illustrated in fig.3.5, and prove asymptotic stability with no velocity feedback. Later on we will show stability of the full model in simulation, and the effects of the tilted rotors on the actual QTRH as seen in experiments. In the case where the lift produced by the four motors (the trajectory of the force on Z_A) is equal to the weight of the robot we get:

$$\ddot{X} = g \tan \theta \quad (5.30)$$

In order to use linear analysis tools assume small θ , giving:

$$\ddot{X} = g\theta \quad (5.31)$$

In the one dimensional case there are no gyroscopic effects, but that is equivalent in real problem to the assumption of soft maneuvers - that is $\Omega_2 \approx \Omega_4 \approx \Omega_1 \approx \Omega_3$, which causes $\tau_{xg}, \tau_{yg}, \tau_{zg}, \tau_{xd}, \tau_{yd}$ from eq. 3.16,3.15 to be approximately zero. Also the gyroscopic effects of the body of the robot itself do not appear in this case which is equivalent to $I_x \approx I_y \approx I_z$. We will define $u_2 = \tau_{ya}$. Now we need to calculate for the one dimension problem τ_{xl} from eq. 3.30. By assigning 3.25 to eq.3.24 and assuming small θ angles, τ_{xl} becomes:

$$\tau_{xl} = -\dot{X}d_l \quad (5.32)$$

With

$$d_l = \frac{1}{2}lcM_{cl}A_d\dot{\Omega}R_b^2\sin\beta \quad (5.33)$$

In this work we will refer to d_l as the velocity momentum coupling coefficient. The dynamics of the pitch becomes:

$$I_y\ddot{\theta} = u_2 + \tau_{xl} = u_2 - \dot{X}d_l \quad (5.34)$$

In order to switch to space state we will define ζ as:

$$\zeta = \begin{bmatrix} \zeta_1 \\ \zeta_2 \\ \zeta_3 \\ \zeta_4 \end{bmatrix} = \begin{bmatrix} \theta \\ \dot{\theta} \\ X \\ \dot{X} \end{bmatrix} \quad (5.35)$$

and the dynamics of the system:

$$\dot{\zeta} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -\frac{d_l}{I_y} \\ 0 & 0 & 0 & 1 \\ g & 0 & 0 & 0 \end{bmatrix} \zeta + \begin{bmatrix} 0 \\ \frac{1}{I_y} \\ 0 \\ 0 \end{bmatrix} u_2 \quad (5.36)$$

As before, we will suggest a control law based on backstepping approach. We will calculate desired pitch angle θ_{ref} according to position error - in this case X . In the previous section we have shown that θ_{ref} has to be also a function of \dot{X} otherwise the system is critically damped and not asymptotically stable. Here we will not use velocity feedback

and show that the system is stable nonetheless.

$$\begin{aligned}
 \theta_{ref} &= -K_{px}X = -K_{px}\zeta_3 \\
 u_2 &= K_{pr}(\theta_{ref} - \theta) - K_{dr}\dot{\theta} = \\
 &K_{pr}(-K_{px}X - \theta) - K_{dr}\dot{\theta} = \\
 &-K'_{px}\zeta_3 - K_{pr}\zeta_1 - K_{dr}\zeta_2
 \end{aligned} \tag{5.37}$$

Assigning u_2 from 5.37 into 5.36 we get:

$$\dot{\zeta} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{-K_{pr}}{I_y} & \frac{-K_{dr}}{I_y} & \frac{-K'_{px}}{I_y} & -\frac{d_l}{I_y} \\ 0 & 0 & 0 & 1 \\ g & 0 & 0 & 0 \end{bmatrix} \zeta \tag{5.38}$$

Correct selection of K_{pr}, K_{dr}, K'_{px} will set all eigenvalues to the left half of the Laplace plain, resulting in a stable system. For example, for $I_y = 1$ and $d_l = 1$ assigning $K_{pr} = 10, K_{dr} = 8, K'_{px} = 1$ results in:

$$\lambda_1 = -6.69$$

$$\lambda_2 = -1.2460$$

$$\lambda_3 = -0.0288 - 1.0944i$$

$$\lambda_4 = -0.0288 + 1.0944i$$

CHAPTER 6

Simulations

6.1. Attitude Regulator

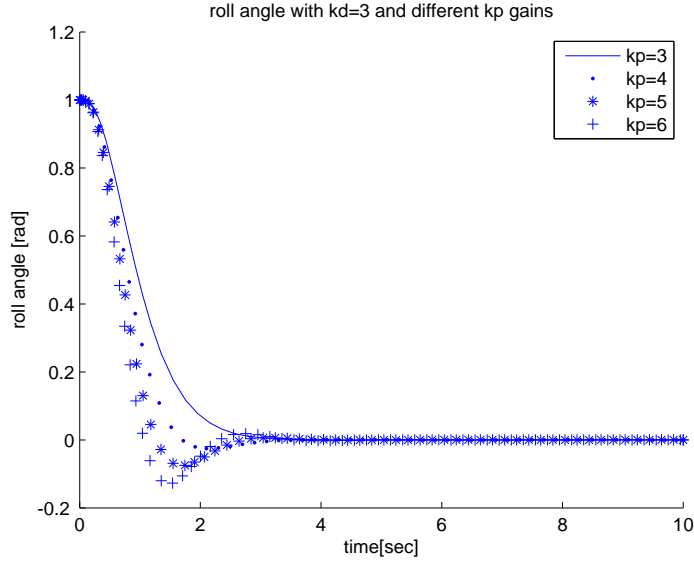
In this section we will present the attitude regulation simulation. The simulations were done using Matlab and Simulink. The quadrotor model used is the one presented in 3.7, propeller time response was also modeled as a first order delay 3.9. The time constant was selected as in [17] to be 0.1 seconds. The mass of the quadrotor was set to $0.5[Kg]$. This mass is consistent with the data measured in our experiment system. In order to estimate moments of inertia, we assumed that each motor weighs $0.06[Kg]$ and neglected the other mass elements. The distance of the motor from the center of mass was set to $0.3[m]$, as in the experimental system. To simulate lack of symmetry, the results were multiplied by a non-symmetry coefficient. This yielded

$$I_x = 0.036 * 0.9[mKg]$$

$$I_y = 0.036[mKg]$$

$$I_z = 0.036 * 2 * 0.8[mKg]$$

The propeller inertia was set to $J = 0.1 * 0.005[mKg]$, according to $10[cm]$ blade length and $0.001[Kg]$ mass, and assuming a uniform mass distribution. The thrust factor b was determined so that the four propellers together produce lift for hover (approximately $5[N]$) in the measured RPM of $4000[RPM]$. This resulted in $b = 0.175\mu[\frac{N}{RPM^2}]$. The

Figure 6.1: PD Regulator with different k_p gains

drag coefficient d was selected to have the same value as b resulting with unity lift to drag ratio.

PD regulator - In figure 6.1 we present the results of the simulation with a simple proportional and differential regulator. $k_d = 3[\frac{N \cdot m \cdot sec}{rad}]$ and $k_p = 3, 4, 5, 6[\frac{N \cdot m}{rad}]$. Initial condition for all variables is zero except for $\phi(0) = 1[rad]$. In figure 6.2 we present the results with $k_d = 3[\frac{N \cdot m \cdot sec}{rad}]$ and $k_p = 3[\frac{N \cdot m}{rad}]$. The initial angles are: $\phi(0) = 1[rad]$, $\psi(0) = -1[rad]$, $\theta(0) = 0.9[rad]$.

This simulation supports section 5.2, where the behaviour of the system under a simple PD controller was analyzed. It is clear that in the simulated scenario this controller stabilizes the system, and that the origin is asymptotically stable. The effects of the coupling between the state variables are visible when comparing fig. 6.2 and fig 6.1. In fig 6.1 convergence is faster and has a smoother curve line for the same gains as in fig 6.2.

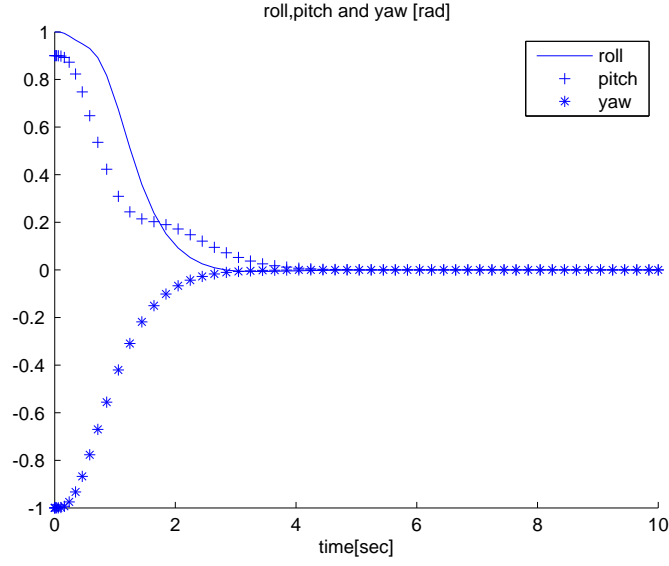


Figure 6.2: PD Regulator roll, pitch and yaw

Feedback Linearization - In this section we will show the performance of the control law presented in 5.16. Using feedback linearization improves the performance of the system by removing the undesired coupling of state variables. Comparing fig.6.3 to fig.6.1 we can see that when using the same gains in both methods, feedback linearization converges faster. Comparing fig. 6.4 to fig 6.2 we can see that under the same initial conditions ($\phi(0) = 1[rad]$, $\psi(0) = -1[rad]$, $\theta(0) = 0.9[rad]$) and gains ($k_d = 3[\frac{N \cdot m \cdot sec}{rad}]$ and $k_p = 3[\frac{N \cdot m}{rad}]$) feedback linearization converges faster than PD controller and with much smoother behaviour. The main drawbacks of the non linear approach is that it requires more input sensors (propeller velocities estimation is needed), and that an accurate model of the system needs to be used, which may be complicated to obtain.

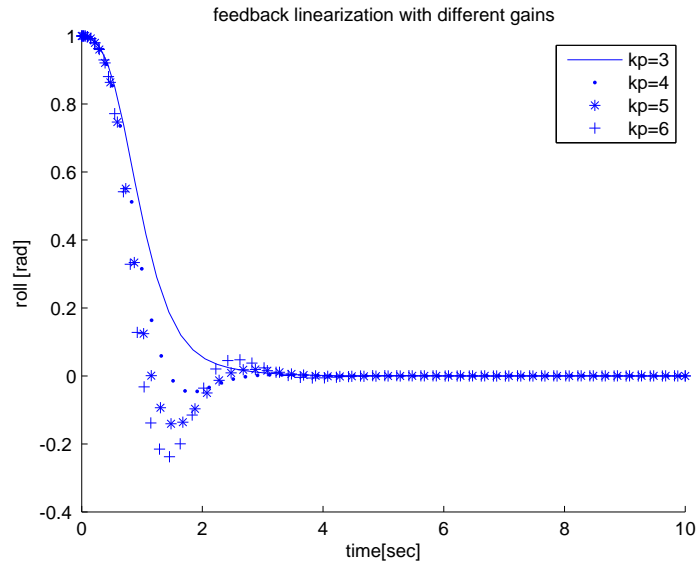


Figure 6.3: feedback linearization regulator with different k_p gains,
 $k_d = 3 \left[\frac{N \cdot m \cdot sec}{rad} \right]$

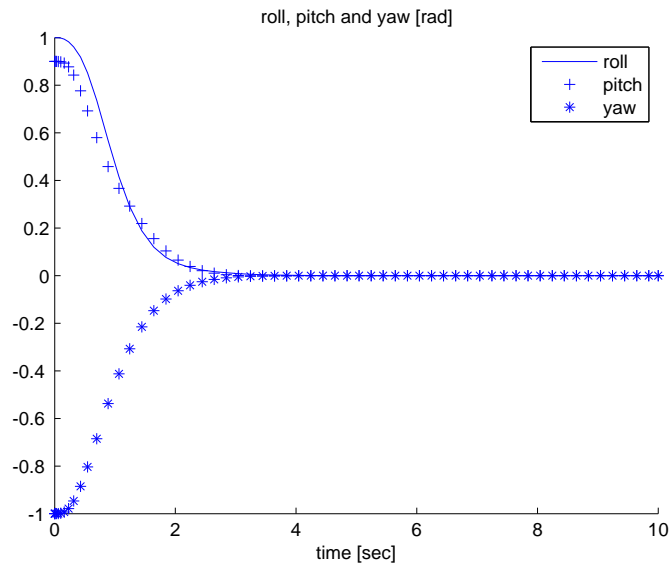


Figure 6.4: feedback linearization regulator roll, pitch and yaw

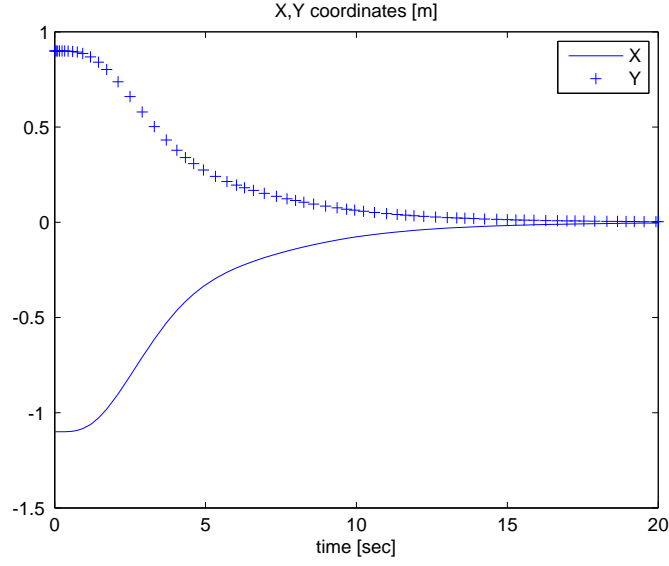


Figure 6.5:

6.2. Position Regulation

In this section we will show the simulation results of a position regulator. The model used for this simulation is the one described in 3.7. For control we used the approach described in section 5.3. Desired yaw was set to zero, and the desired roll and pitch were set using a simple PD controller. Pitch was set according to error along the X axis, and yaw according to error along the Y axis. Attitude control was the law used in feedback linearization 5.16. Gains were selected experimentally to be $K_p = 7[\frac{N \cdot m}{rad}]$, $k_d = 5[\frac{N \cdot m \cdot sec}{rad}]$. Gains chosen for the outer control loop were - $K_p = 0.6[\frac{deg}{m}]$ and $K_d = 2.7[\frac{deg \cdot sec}{m}]$. The results for starting conditions $X(0) = -1.1[m]$ and $Y(0) = 0.9[m]$ are presented in fig.6.5. The outer control loop calculates the desired roll and pitch angles, and an inner control loop calculates the actual control signals u_2, u_3 and u_4 . The quality of attitude tracking is presented for

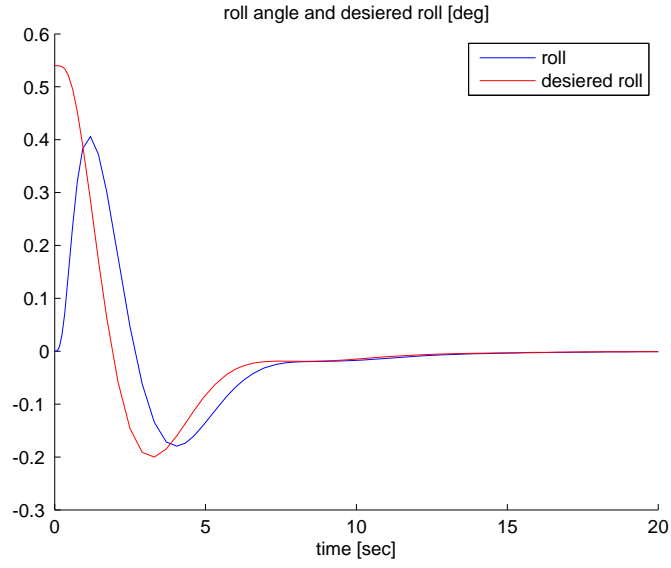


Figure 6.6:

the roll angle at fig.6.6 As described in section 5.3, since there is no drag in the model, damping is crucial for stability. In fig 6.7 we show the results of the same simulation but with a smaller K_d gain for position - $0.3[\frac{deg \cdot sec}{m}]$. The system becomes unstable.

6.3. QTRH Simulations

In this section we will present the results of the QTRH simulation. The model used was a simplified version of the modeled developed at chapter 3. This model is valid under the assumptions of small yaw

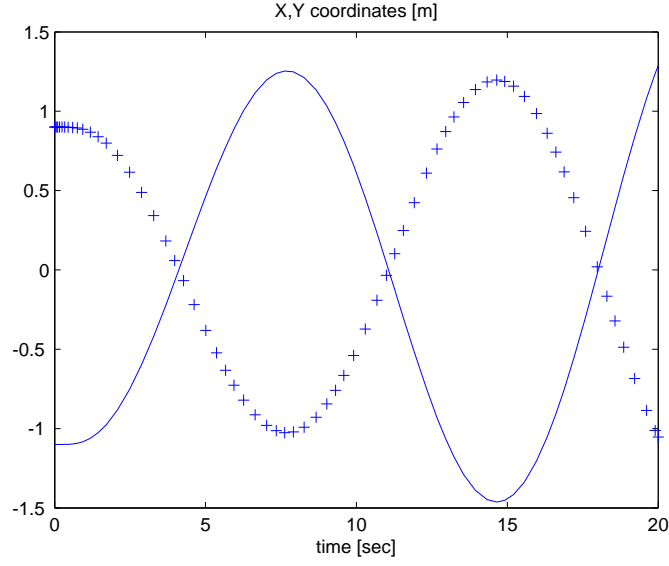


Figure 6.7: Unstable position regulation due to small K_d - $K_d = 0.3 \frac{\text{deg} \cdot \text{sec}}{\text{m}}$

angles and small velocity along the Z axis:

$$\begin{cases} I_x \ddot{\phi} = \dot{\theta} \dot{\psi} (I_y - I_z) + u_2 - J \dot{\theta} \Omega_r + d_l \dot{Y} \\ I_y \ddot{\theta} = \dot{\phi} \dot{\psi} (I_z - I_x) + u_3 + J \dot{\phi} \Omega_r - d_l \dot{X} \\ I_z \ddot{\psi} = \dot{\phi} \dot{\theta} (I_x - I_y) + u_4 \\ \ddot{Z} = -g + (c\phi c\theta) \frac{u_1}{m} \\ \ddot{X} = (c\phi s\theta c\psi + s\phi s\psi) \frac{u_1}{m} \\ \ddot{Y} = (c\phi s\theta s\psi - s\phi c\psi) \frac{u_1}{m} \end{cases} \quad (6.1)$$

d_l is the velocity momentum coupling coefficient 5.33. When repeating the same simulation as in previous section, where the results were unstable and shown in fig.6.7, only this time using the QTRH model 6.1 with $d_l = 0.2 [N \cdot \text{sec}]$, we see that the QTRH is stable and converges to the origin (fig.6.8). This simulation shows that the QTRH position regulator remains stable with a lower velocity feedback gain. In fact,

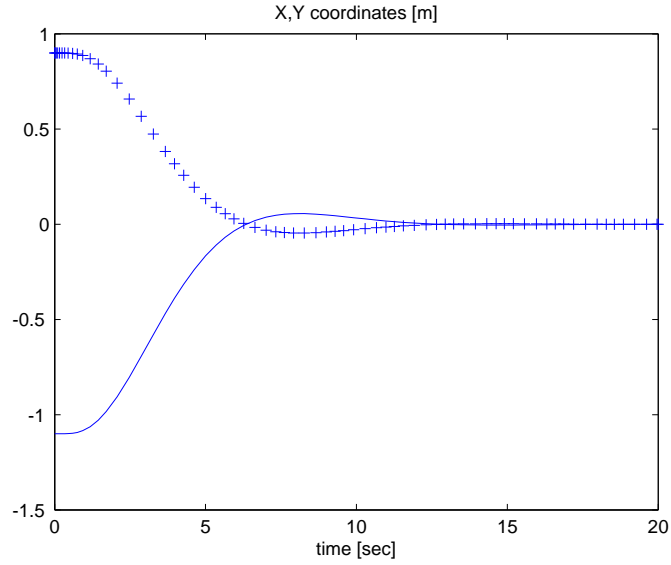


Figure 6.8: QTRH simulation with same gains as in fig.6.7 and $d_l = 0.2[N \cdot sec]$

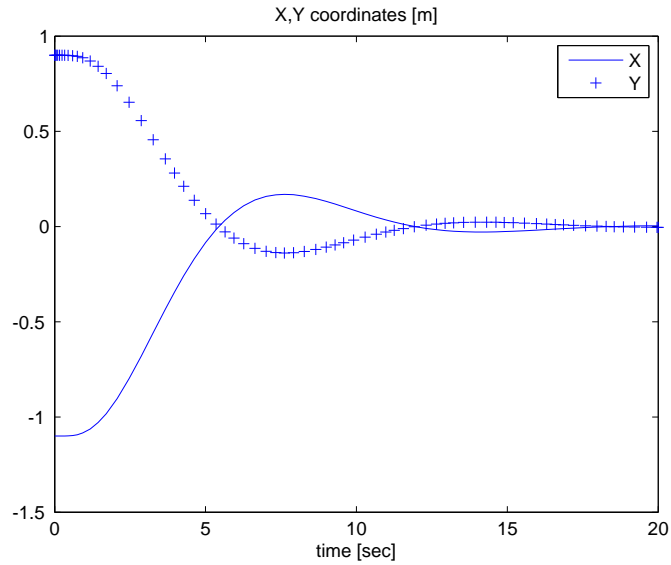


Figure 6.9: QTRH simulation with $k_d = 0$ and $d_l = 0.2[N \cdot sec]$

the regulator will remain stable with no velocity feedback at all, as seen in fig 6.9. There for the same K_p and d_l as before, we used $k_d = 0$, and as we can see the system was still stable.

CHAPTER 7

Experiments

In this chapter we will discuss the results of the experiments we conducted to validate the theory presented in the previous chapters. We have built several quadrotor robots using different materials for the structure itself. The first quadrotor was made out of carbon fiber plates, which were very elastic (fig. 7.1). It turned out that the elasticity caused severe vibrations which interfered with the sensor readings. Later on, the structure was replaced by a new structure made of carbon

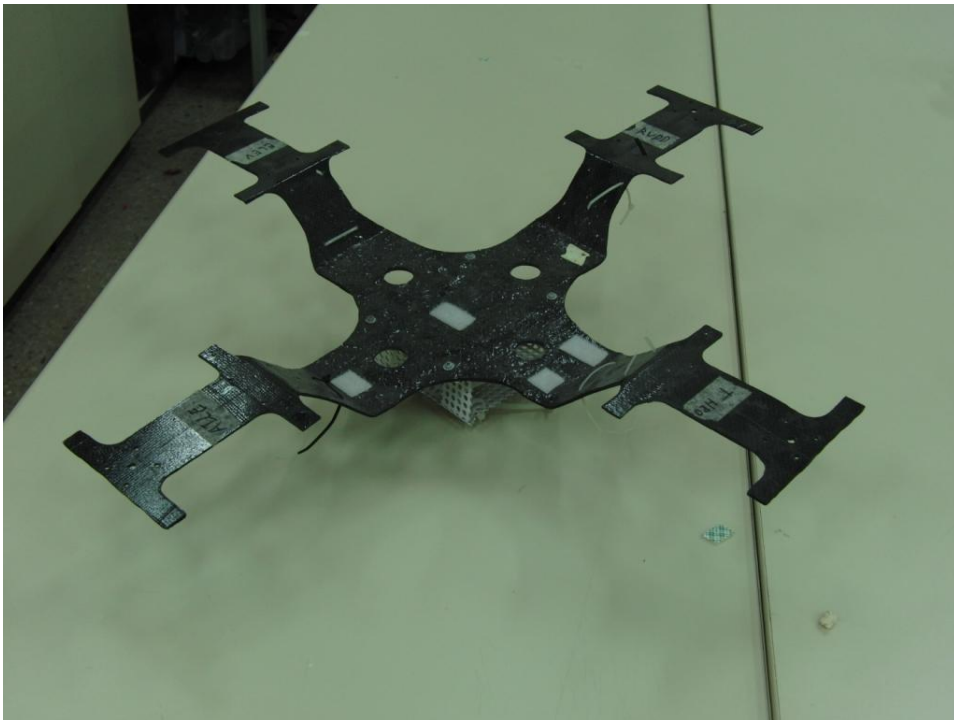


Figure 7.1: First quadrotor constructed in our lab

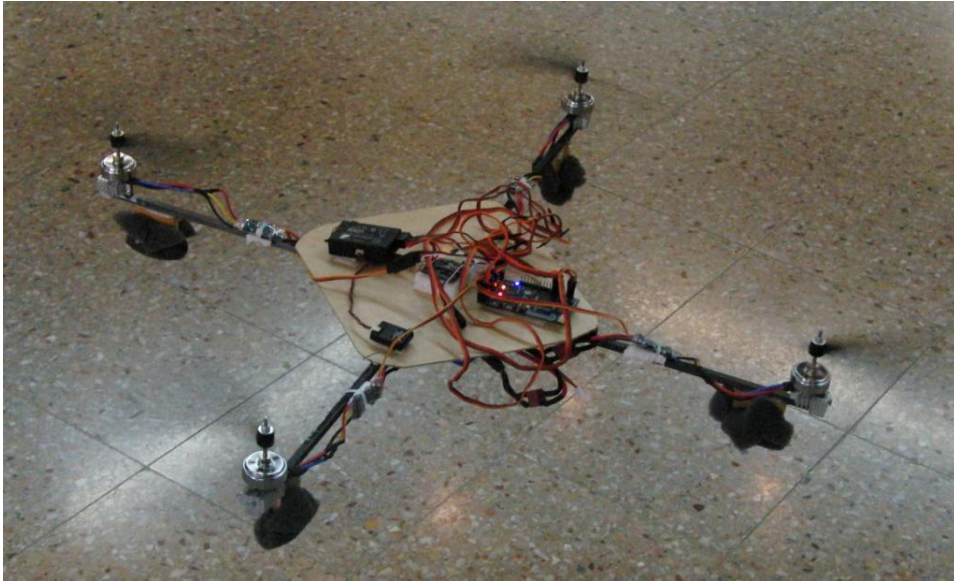


Figure 7.2: Second quadrotor constructed in our lab

fiber rods(fig. 7.2). This structure proved to be much more resistant to vibrations a stable flight was achieved. However, the structure was not resistant enough in crashes and the carbon fiber rods were later replaced with aluminum rods. On board the quadrotor was a sensor board, including a three axis accelerometer and a three axis gyro sensors. All processing including orientation estimation and control, was done on-board using an Atmega1280 eight bit microprocessor. The power source chosen was LiPo 3 cell battery pack. The motors used were outrunner brushless small scale motors, spinning eight inch long propellers with 3.8 inch pitch.

7.1. Attitude Estimation Experiment

The heart of the quadrotor stabilization system is the IMU. The performance of the quadrotor is greatly effected by the quality of state

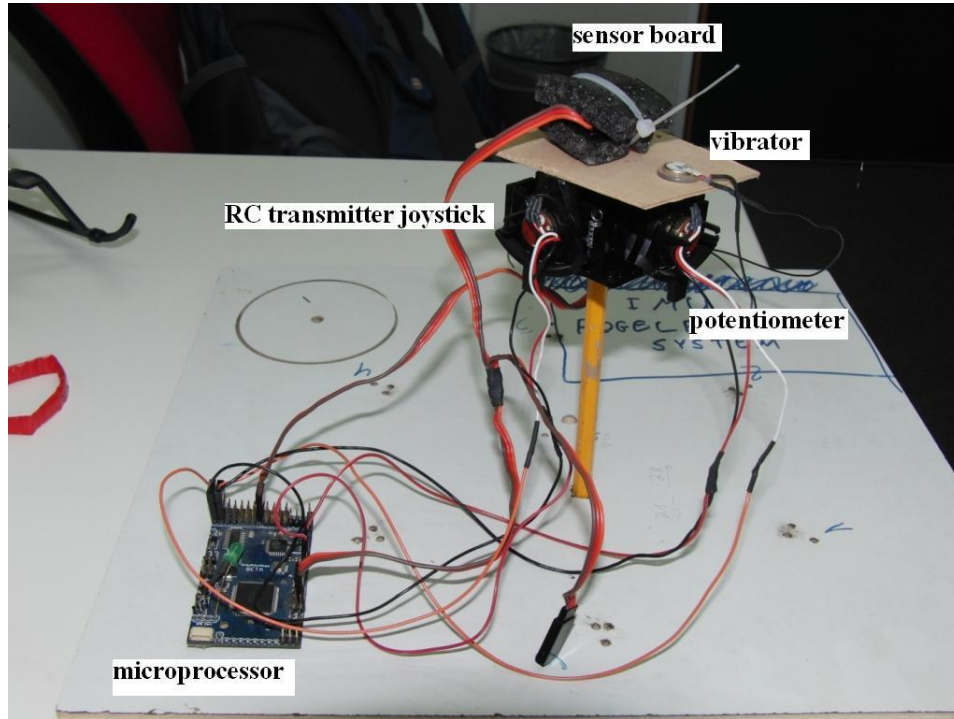


Figure 7.3: IMU evaluation system

estimation. Our first step was to construct an experiment system that can validate the quality of the pitch and roll estimation.

The experiment system, fig 7.3, is composed of a sensor board including three axis accelerometer and three axes gyro sensors, a joystick dismounted off of a RC transmitter which includes two potentiometers, and a microprocessor board. A vibrator was also installed to examine the effects of vibrations. The sensors are used by the microprocessor as an input for the estimation algorithm, and the results are compared with the potentiometers readings, which are considered as a direct and accurate measurement. In figure 7.4 we see the results of the algorithm presented in 4.2 when applying different α values. When using higher α values the algorithm relies more on the gyros, thus errors accumulated

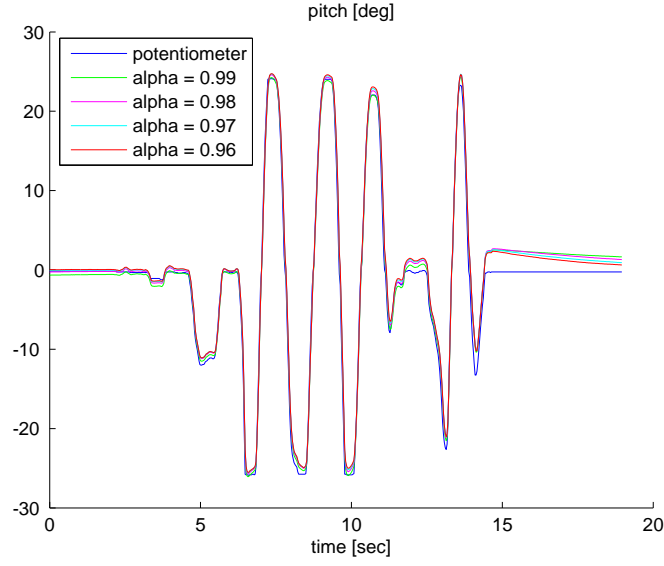


Figure 7.4: IMU evaluation system

in integration and numeric calculations are corrected relatively slowly. On the other hand, when using low values of α , the algorithm becomes more sensitive to vibrations, as seen in fig.7.5.

7.2. Accelerometers in Attitude estimation

In this experiment we have created a data log containing raw accelerometer sensor reading during an aggressive flight around a fixed point. The goal of the experiment is to validate the theory presented in section 4.4, which claims that no data about orientation can be obtained from the accelerometers during a hover under the assumption of no drag (as seen in 4.28). In other words, this experiment examines whether or not the effects of drag on the estimation algorithm, while in a state of a near-hover, are substantial or not. The raw measurements are seen in fig. 7.7. Take off took place after 25 seconds. The vibrations

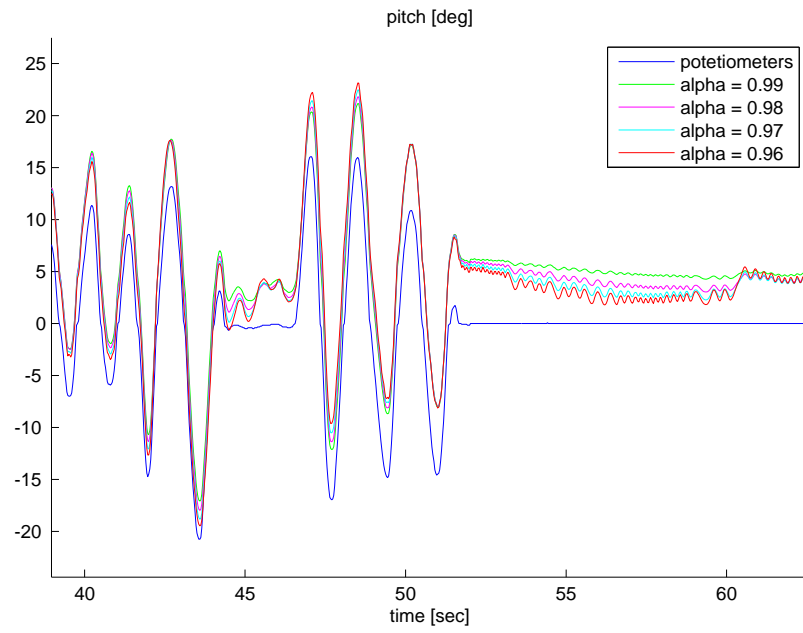


Figure 7.5: IMU evaluation system - pitch measurements and calculations under vibrations.

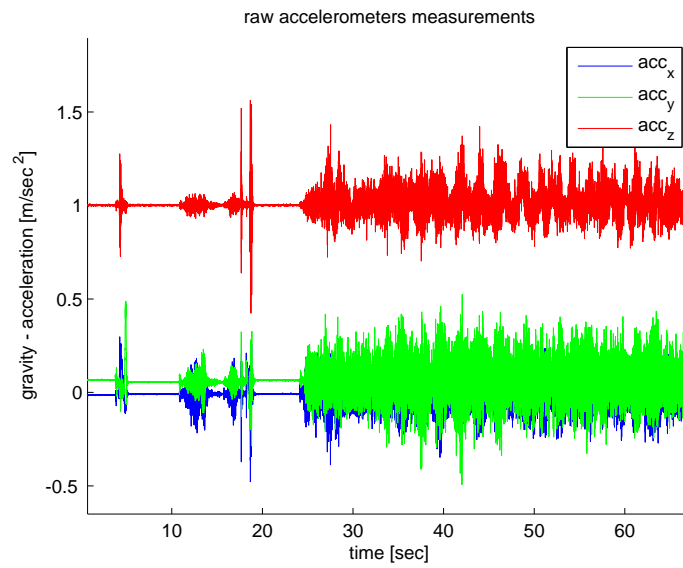


Figure 7.6: Raw acceleration data during a near-hover state

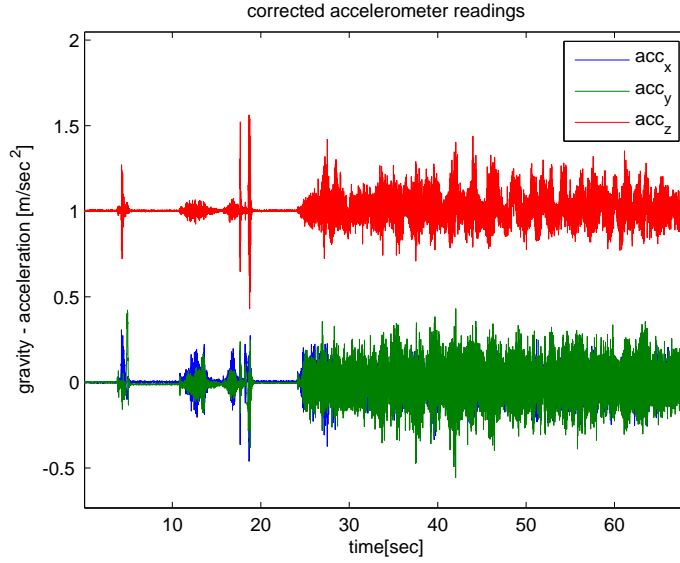


Figure 7.7: Raw acceleration data during a near-hover state

caused by the propellers are very clear in the graphs. Miss-alignment of the sensors is visible and a correction rotation matrix is calculated, in order to correct the measurements from clean samples before the take off:

$$acc_{corrected} = R_{correction} \cdot acc \quad (7.1)$$

The corrected measurements are presented in fig.7.7. The flight took place indoors in a small space - hence a near-hover state. The pilot performed relatively aggressive maneuvers while flying in straight lines. A diagram of roll and pitch angles is presented in fig. 7.8. When used in a complementary filter, the corrected accelerometer measurements are filtered as described in 4.13. In fig. 7.9, the filtered accelerometer data is shown. We chose a relatively high frequency LPF with $\alpha = 0.98$. This number was selected based on good flight performance achieved in experiments with the algorithm presented in section 4.2. Changes in acc_x are very small even though the robot flew in high angles. A

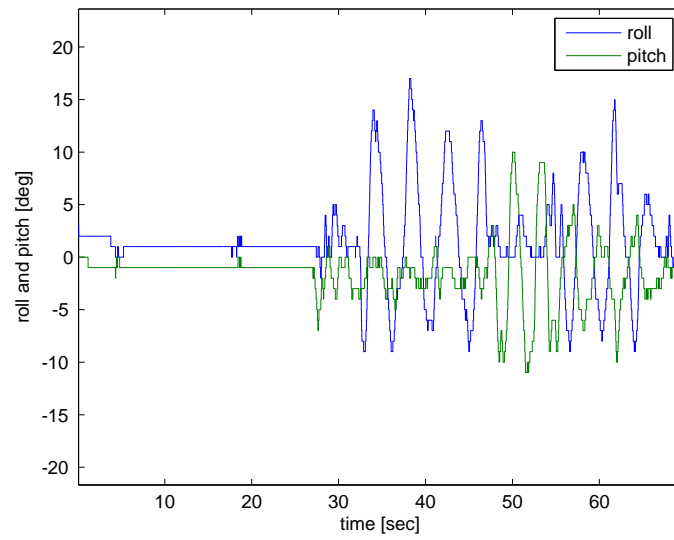


Figure 7.8: Roll and pitch angles during the flight

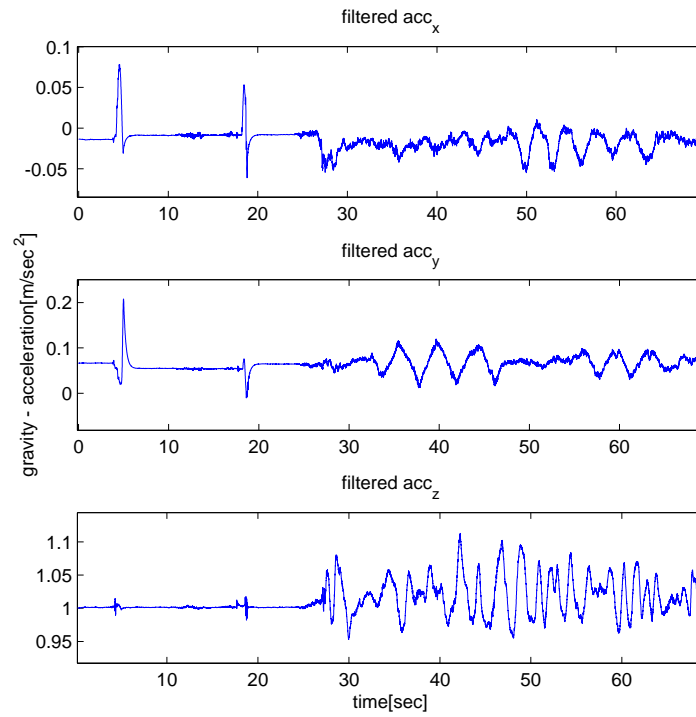


Figure 7.9: Accelerometer data after filtering

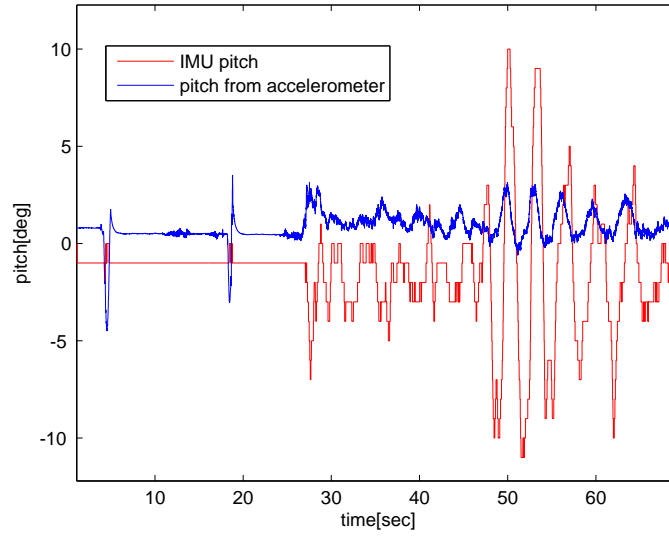


Figure 7.10: pitch from IMU compared with pitch from accelerometers

comparison between the pitch calculated with The IMU and from the filtered accelrometer as in 4.3 is shown in fig.7.10

Conclusion - some correlation between the angles calculated by the IMU and the accelerometer exists due to the effects of drag. However, in our opinion, due to the distortions caused by the filtering and due to other interference, such as signals generated from acceleration caused by external forces, the use of accelerometers in orientation estimation during a hover is not recommended. Drift correction should be achieved by a different direct measurement of orientation or by monitoring the location of the robot, using a GPS receiver or a similar method.

7.3. Position Regulator

In this section we will show the experimental results of a position regulator. We used the approach described in section 5.3. Desired

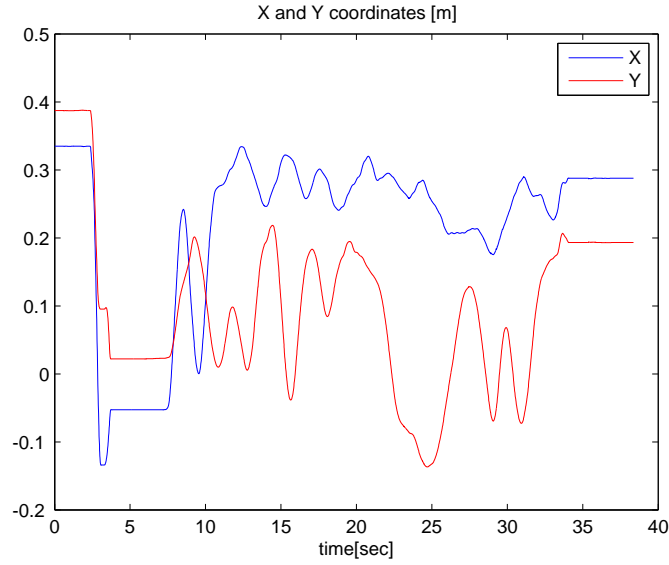


Figure 7.11: Position regulator results

yaw was set to zero, and the desired roll and pitch were set using a simple PD controller. Pitch was set according to error along the X axis and yaw according to error along the Y axis. Fig 7.11 shows the results of the regulator. In that case we used the following gains - $K_p = 7[\frac{deg}{m}]$, $K_d = 5[\frac{deg \cdot sec}{m}]$. The system is stable, but position is not completely fixed and there is a steady state error. This is caused by relatively poor attitude control due to slow processing time, and noisy sensor readings. The steady state error is caused by the lack of an integral component in the control law, coupled with asymetry and alignment error of the sensors. Nonetheless, the results can be consider sufficient for most applications.

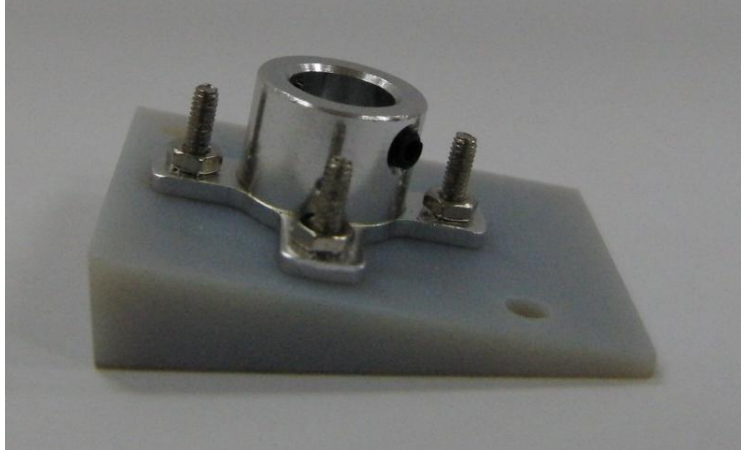


Figure 7.12: Motor tilting Mechanism

7.4. QTRH Experiments

In this section we will show the effects of the tilted motor mechanism. For the experiments we used four plastic prism-shaped devices manufactured with a 3D printer (fig. 3.4, 7.12). The tilting angle achieved using these devices is $\beta = 15[deg]$. This plastic devices can be removed thus making the robot a regular quadrotor. The idea was to show that without velocity feedback the quadrotor cannot achieve a stable hover, as apposed to the QTRH. However, as seen in fig. 7.13, the drag was substantial enough to damp the system. Because of this we conducted a different experiment in which we increased the proportional gain until the regular quadrotor became unstable - fig 7.14. Then we examined the QTRH at the exact same gains - $K_d = 0, K_p = 15[\frac{deg}{m}]$. The results are shown in fig. 7.15.

Conclusion. In fig.7.15 we see that the QTRH is stable but critically damped. The reason for this is that the coupling between linear velocity and angular momentum is too weak to damp the system. In

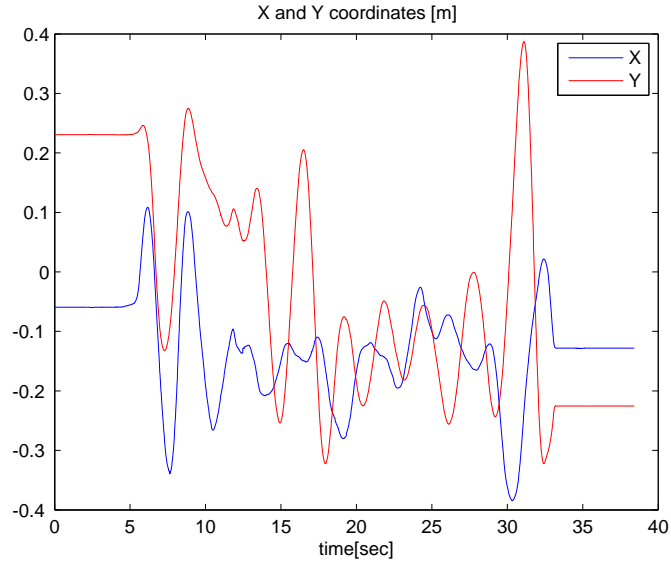


Figure 7.13: Position regulator results with no velocity feedback, $K_p = 7[\frac{deg}{m}]$

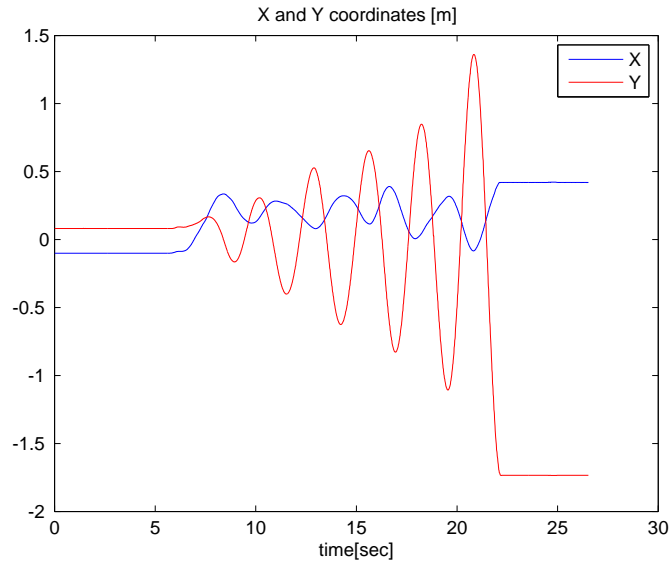


Figure 7.14: Unstable position regulator results - $K_p = 15[\frac{deg}{m}]$, $K_d = 0$

other words d_l is too small. This is supported by other experiments which showed that the effect of the motor tilting is not sufficient to justify the enlarged energy consumption of the QTRH. On the other

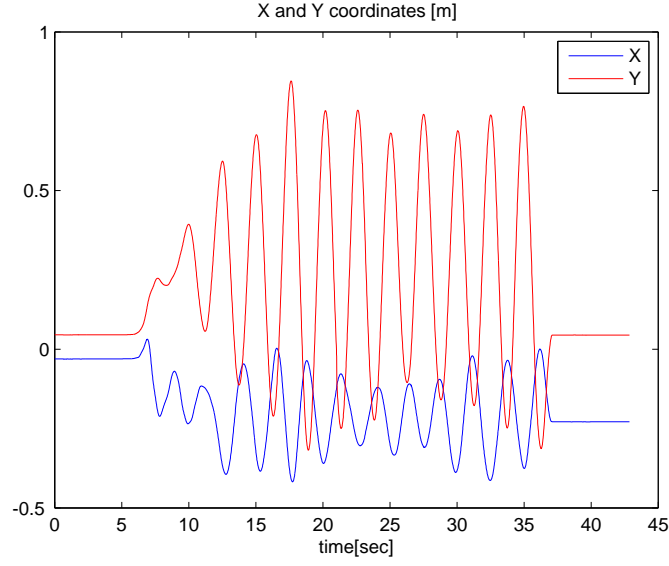


Figure 7.15: Unstable position regulator results - $K_p = 15[\frac{deg}{m}]$, $K_d = 0$

hand the experiment showed clearly that the damping effect exists, and can be used. In our opinion, in order to justify the energy loss the rotors should be tilted only in applications where damping of linear inertia is critical, and no accurate velocity measurement is available. In these cases d_l should be increased, by using large propellers, rather than increasing β to avoid additional energy loss. This mechanism will be also useful in larger scale quadrotors, where the natural damping effect caused by drag is less substantial.

CHAPTER 8

Conclusions and Future Work

Estimation and control -

1 - During our work, we came to the conclusion that simple linear controllers are adequate and yield good performance. For most applications a more complex control law is not necessary. On the other hand, estimating the state of the quadrotor is a complicated problem which is not yet completely solved. In our opinion, this is partially due to insufficient information acquired from the sensors. For this reason we suggest in future work to study the integration of visual feedback in the existing estimation algorithms. For example linear transformations, such as two dimensional Fourier transforms and Hough transforms, applied to an image taken with a camera mounted on an aircraft, parallel to the aircrafts roll axis, can be used for direct estimation of the roll angle. An example is given in [9.2](#).

2 - Since the linear velocity of the quadrotor is affected by the roll and pitch angles we also suggest researching an estimator that will use GPS position and velocity measurements to estimate gyro bias.

3 - One of the reasons that during this work we preferred simple PD control laws, is the difficulty in acquiring an accurate model with accurate parameters. For this reason we suggest researching the use of

the discrete model 3.13 with different adaptive parameter estimation algorithms for real time model estimation.

Dynamic dumping -

The dynamic damping concept we presented here needs to be validated experimentally on a real flying platform. The most suitable platforms are those which use internal combustion engines, due to the strong vibration.

Thermopiles in attitude estimation -

The mechanism presented in fig. 4.2 should be built and tested and the effects of different servo controllers on the quality of estimation under different flight characteristics should be studied. This mechanism offers substantial advantages for outdoor, high altitude applications.

QTRH -

The special structure of the QTRH does improve its stability but at a cost of decreased energetic efficiency. More experiments should take place with different parameters. Theoretically, the effects should be more critical in heavier quadrotors or when using faster motors. During this work we designed and built a platform that can change the angles of the QTRH during flight, thus cancelling the energy loss from the tilted motors when the extra damping is not required, as in the case of flying in a constant speed. This mechanism will also allow to control the magnitude of the mechanical velocity feedback mid-flight. This is

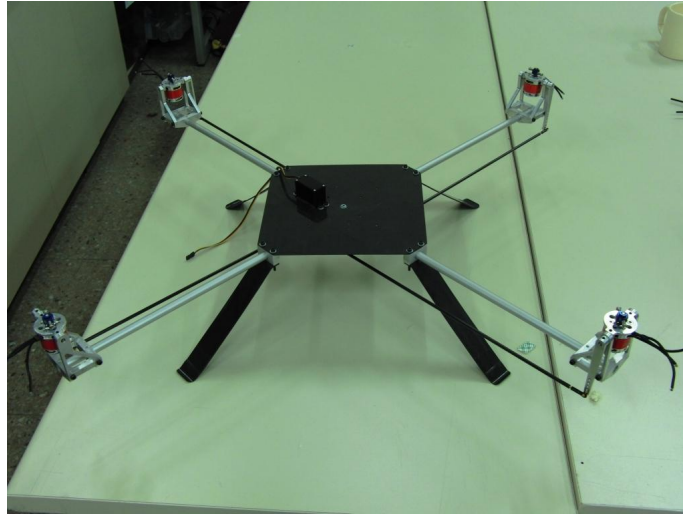


Figure 8.1: Variable β QTRH

still work in progress and more experiments will take place soon. Fig. 8.1 shows the structure itself, made of carbon fiber and aluminum. Fig. 8.2 shows the motor mount. The motor is connected to a hinge and is capable of being positioned in a range of angles. A carbon fiber rod is connected to the mechanism to generate the required moment. Fig. 8.3 shows the servo mechanism that controls the position of all the four motors.

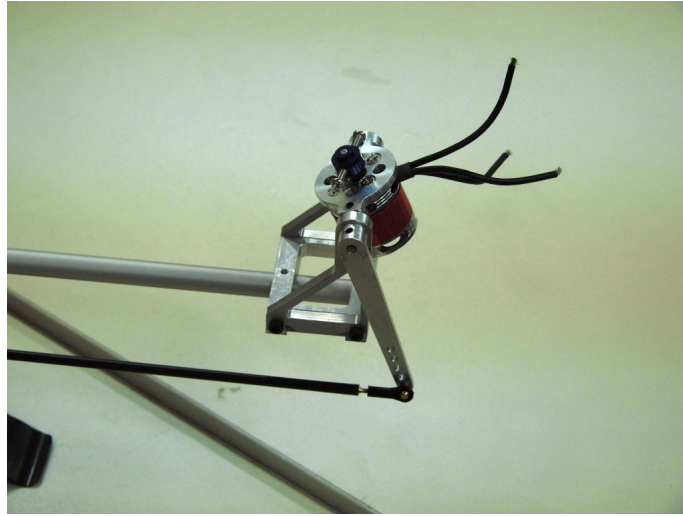


Figure 8.2: Variable β QTRH motor mount

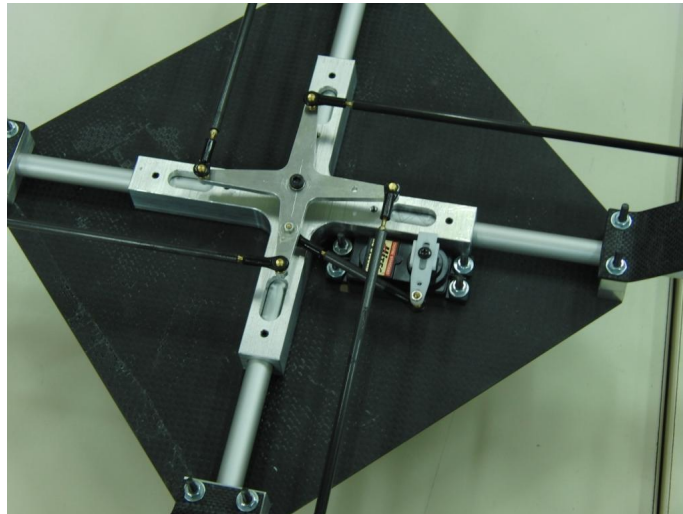


Figure 8.3: Variable β QTRH servo mechanism

CHAPTER 9

Apendix

9.1. Discrete quadrotor model

Mathematical Derivation. The laplace transform of the open-loop system from fig.3.2 is given by the multiplication of H_{ZOH} 3.12 and the Transfer function of the quadrotor robot attitude 3.11:

$$\frac{\Phi}{U_{2ref}} = \frac{l}{I_x} \frac{1}{(\tau_d s + 1)s^3} (1 - e^{-sT}) \quad (9.1)$$

To obtain the equivalent discrete model We need to rearrange the model so we can use conversion tables from Laplace domain to z domain by splitting the expression into partial fractions:

$$\frac{\Phi}{U_{2ref}} = \left(\frac{l}{I_x s^3} - \frac{l\tau_d}{I_x s^2} + \frac{l\tau_d^2}{I_x s} - \frac{l\tau_d^3}{I_x(\tau_d s + 1)} \right) (1 - e^{-sT}) \quad (9.2)$$

multiplication in Laplace domain by $(1 - e^{-sT})$ is equivalent to multiplication by $(1 - z^{-1})$ in the z domain. Now it is simple to convert each of the fractions using a conversion table ([10] - appendix 5). The resulting Transfer function:

$$\frac{\Phi}{U_{2ref}} = \frac{l}{I_x} \left(\frac{T^2 z(z+1)}{2(z-1)^3} - \frac{\tau_d T z}{(z-1)^2} + \frac{\tau_d^2 z}{z-1} - \frac{\tau_d^2 z}{z - e^{-T/\tau_d}} \right) (1 - z^{-1}) \quad (9.3)$$

Which results in 3.13



Figure 9.1: Common urban image.

9.2. Attitude estimation based on visual feedback

Figures 9.1, 9.3, 9.5, show a scenery image taken in an urban environment, and the same image rotated by 22 and 44 degrees. Figures 9.2, 9.4, 9.6 show the corresponding two dimensional discrete Fourier transforms. Since most energy in the image is in the vertical and horizontal spacial frequencies it is possible to extract the rotation amount around the focal axis by determining the direction of the spacial frequencies with the most energy. Hough transforms can be used for this purpose.

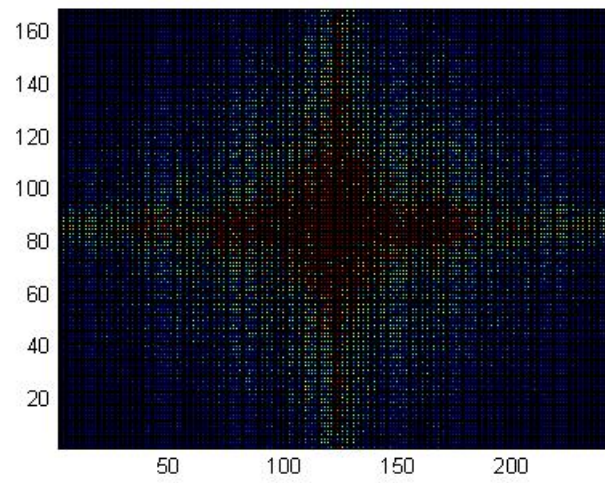


Figure 9.2: Fig 9.1 two dimensional Discrete Fourier Transform



Figure 9.3: The same scenery as in 9.1, rotated in 22 degrees

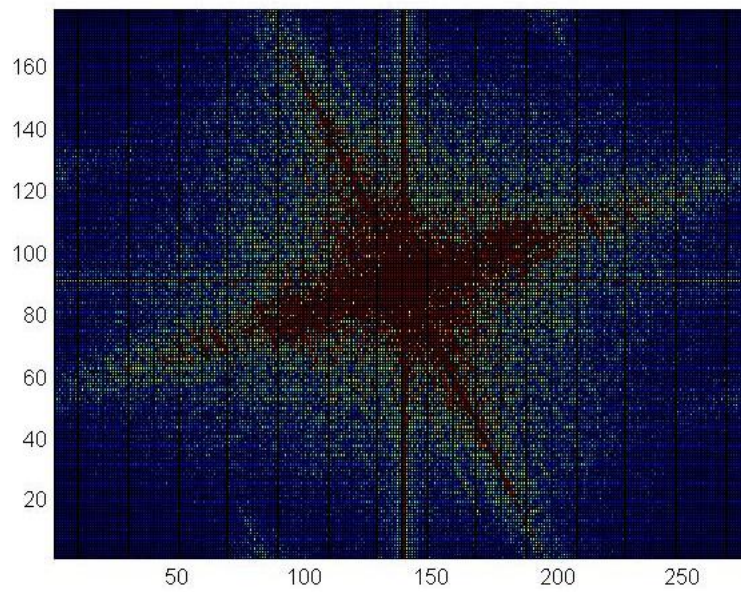


Figure 9.4: Fig 9.3 two dimensional Discrete Fourier Transform



Figure 9.5: The same scenery as in 9.1, rotated in 22 degrees

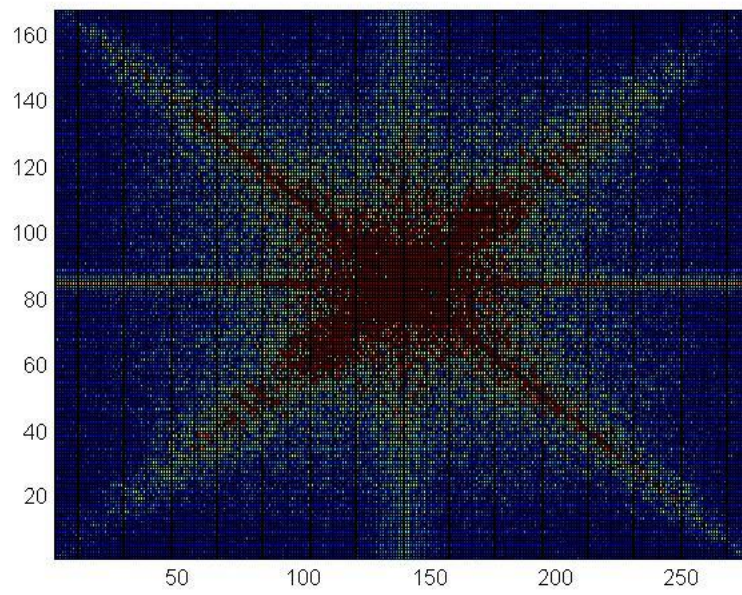


Figure 9.6: Fig 9.5 two dimensional Discrete Fourier Transform

Bibliography

- [1] <http://www.draganfly.com/uav-helicopter/draganflyer-x6/>.
- [2] <http://www.mikrokoetter.de/ucwiki/en/MikroKopter>.
- [3] <http://www.rctoys.com/rc-toys-and-parts/DF-VTI/RC-HELICOPTERS.html>.
- [4] E. Altug, J.P. Ostrowski, and C.J. Taylor. Quadrotor control using dual camera visual feedback. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, volume 3, pages 4294–4299. IEEE, 2003.
- [5] KJ Astrom, CC Hang, P. Persson, and WK Ho. Towards intelligent pid control* 1. *Automatica*, 28(1):1–9, 1992.
- [6] Samir BOUABDALLAH. *design and control of quadrotors with application to autonomous flying*. PhD thesis, EPFL, 2007.
- [7] Samir Bouabdallah and Roland Siegwart. Backstepping and sliding-mode techniques applied to an indoor micro quadrotor. *International Conference on Robotics and Automation Barcelona, Spain*, 2005.
- [8] Cosmin Coza and C.J.B. Macnab. A new robust adaptive-fuzzy control method applied to quadrotor helicopter stabilization. 2006.
- [9] F. Croce, L. Persi, F. Ronci, and B. Scrosati. Nanocomposite polymer electrolytes and their impact on the lithium battery technology. *Solid State Ionics*, 135(1-4):47–52, 2000.
- [10] K. Dutton. *The art of control engineering*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1997.
- [11] C. Edwards and S.K. Spurgeon. *Sliding mode control: theory and applications*. Taylor & Francis systems and control book series. Taylor & Francis, 1998.
- [12] S.G. Fowers, D.J. Lee, B.J. Tippetts, K.D. Lillywhite, A.W. Dennis, and J.K. Archibald. Vision aided stabilization and the development of a quad-rotor

- micro uav. In *Computational Intelligence in Robotics and Automation, 2007. CIRA 2007. International Symposium on*, pages 143–148. IEEE, 2007.
- [13] Samuel Fux. *Development of a planar low cost Inertial Measurement Unit for UAVs and MAVs*. PhD thesis, ETH, 2008.
 - [14] S. Waslander G. Hoffmann and C. Tomlin. Quadrotor helicopter trajectory tracking control. *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2008.
 - [15] P.R. Gray, R.G. Meyer, P.J. Hurst, and S.H. Lewis. Analysis and design of analog integrated circuits. 1993.
 - [16] WT Higgins. A comparison of complementary and kalman filtering. *Aerospace and Electronic Systems, IEEE Transactions on*, (3):321–325, 1975.
 - [17] G.M. Hoffmann, S.L. Waslander, and C.J. Tomlin. Quadrotor helicopter trajectory tracking control. *Electrical Engineering*, (August):1–14, 2008.
 - [18] D.F. Hornig and BJ O’keefe. The design of fast thermopiles and the ultimate sensitivity of thermal detectors. *Review of Scientific Instruments*, 18(7):474–482, 1947.
 - [19] A. Hughes. *Electric motors and drives: fundamentals, types and applications*. Newnes, 2006.
 - [20] K. Iizuka, H. Uzuhashi, M. Kano, T. Endo, and K. Mohri. Microcomputer control for sensorless brushless motor. *Industry Applications, IEEE Transactions on*, (3):595–601, 1985.
 - [21] P.A. Ioannou and J. Sun. Robust adaptive control. In *American Control Conference, San Diego, California, USA*, 1984.
 - [22] N.G. Johnson. *Vision-assisted control of a hovering air vehicle in an indoor setting*. PhD thesis, Masters thesis, BYU, 2008.
 - [23] R.E. Kalman et al. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.
 - [24] H.K. Khalil. *Nonlinear systems*. Prentice Hall, 2002.
 - [25] M. Koifman and IY Bar-Itzhack. Inertial navigation system aided by aircraft dynamics. *Control Systems Technology, IEEE Transactions on*, 7(4):487–493, 1999.

- [26] J.B. Kuipers. *Quaternions and rotation sequences*. Princeton university press Princeton, NJ, 1999.
- [27] Robert Mahony Jonghyuk Kim Mark Euston, Paul Coote and Tarek Hamel. A complementary filter for attitude estimation of a fixed-wing uav.
- [28] P.S. Maybeck. *Stochastic models, estimation and control*, volume 141. Academic press, 1979.
- [29] D. Mellinger, N. Michael, and V. Kumar. Trajectory generation and control for precise aggressive maneuvers with quadrotors. In *Proceedings of the International Symposium on Experimental Robotics*, Dec 2010.
- [30] R.M. Murray, Z. Li, and S.S. Sastry. *A mathematical introduction to robotic manipulation*. CRC, 1994.
- [31] K.S. Narendra and A.M. Annaswamy. *Stable adaptive systems*. Prentice-Hall, Inc., 1989.
- [32] H.G. Nguyen, J. Morrell, K. Mullens, A. Burmeister, S. Miles, N. Farrington, K. Thomas, D.W. Cage, SPACE, and NAVAL WARFARE SYSTEMS CENTER SAN DIEGO CA. *Segway robotic mobility platform*. Citeseer, 2004.
- [33] SR Ovshinsky, MA Fetcenko, and J. Ross. A nickel metal hydride battery for electric vehicles. *Science*, 260(5105):176, 1993.
- [34] Jean-Michel Pflimlin Robert Mahony, Tarek Hamel. Complementary filter design on the special orthogonal group $so(3)$. 2005.
- [35] Tarek Hamel Robert Mahony and Jean-Michel Pflimlin. Nonlinear complementary filters on the special orthogonal group. *IEEE TRANSACTIONS ON AUTOMATIC CONTROL*, 53, 2008.
- [36] Tarek Hamel Robert Mahony, Sung-Han Cha. A coupled estimation and control analysis for attitude stabilisation of mini aerial vehicles. 2006.
- [37] Angelo M. Sabatini. Quaternion-based extended kalman filter for determining orientation by inertial and magnetic sensing. *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING*, 53, 2006.
- [38] Andk Noth Samir Bouabdallah and Roland Siegwan. Pid vs lq control techniques applied to an weight augmentation high energy conrunption indoor micro quadrotor. *Intelligent Robots and Systems ,Sendal, Japan*, 2004.

- [39] S. Sastry and M. Bodson. *Adaptive control: stability, convergence, and robustness*. 1989.
- [40] John L Titterton, David H.; Weston. *Strapdown inertial navigation technology*. AIAA.
- [41] G.P. Tournier, M. Valenti, J.P. How, and E. Feron. Estimation and control of a quadrotor vehicle using monocular vision and moire patterns. In *AIAA Guidance, Navigation and Control Conference and Exhibit*, pages 21–24. Citeseer, 2006.
- [42] CY Tzeng, GC Goodwin, and S. Crisafulli. Feedback linearization design of a ship steering autopilot with saturating and slew rate limiting actuator. *International journal of adaptive control and signal processing*, 13(1):23–30, 1999.
- [43] V.I. Utkin. Sliding mode control. *Variable structure systems: from principles to implementation*, 66:1, 2004.
- [44] A. Visioli. *Practical PID control*. Springer Verlag, 2006.
- [45] G. Welch and G. Bishop. An introduction to the kalman filter. *University of North Carolina at Chapel Hill, Chapel Hill, NC*, 7(1), 1995.

תקציר

מטרת עבודה זו היא חקר ההיבטים המרכזיים בתכנון ופיתוח של רחפן אוטונומי בעל ארבעה מנועים. היבטים אלו כוללים מבנה, מידול, בקרה ושיערוך מצב של הרחפן. נסקור מודלים דינאמיים שונים, ושיטות בקרה שונות לייצוב. בתחום המבנה והמידול נציג מבנה חדשני של רחפן בעל מנועים מוטים, אשר נותן למערכת את התכונה הרצויה של משוב שלילי למהירות הקווית באופן טבעי, ובכך משפר את יציבות המערכת ומקטין את התלות במדידת מהירות מדויקת, שהיא לעיתים קשה להשגה. נעזר בסימולציות וניסויים כדי להראות את קיום המשוב הטבעי. בתחום השערוך נסקור את השימוש בחיישנים השונים ונציג שיטות שונות למיזוג המידע מהם לשערוך ווקטור המצב של המערכת. נציג שיטה חדשה לשימוש בחיישני קרינת גוף שחור למדידה ישירה של זווית הגלגול והעלרוד. כמו כן נציג שיטה חדשה לשיפור המדידות הנעשות בחיישני התאוצה הנפוצים ע"י שיכוך דינאמי של רעידות. בתחום הבקרה נשתמש בגרפים תלת ממדיים כדי להראות את השפעת הוספת משוב תאוצה זוויתית. נציג חוקי בקרה שונים לייצוב ברחפן הקלאסי, ולייצוב הרחפן בעל המנועים המוטים. נראה כי כאשר משתמשים במבנה בעל מנועים מוטים קטן הצורך במשוב חיצוני למהירות הקווית להשגת יציבות במיקום.



אוניברסיטת בן-גוריון בנגב

הפקולטה למדעי ההנדסה

המחלקה להנדסת מכונות

מבנה, שיערוך מצב ובקרה של רחפן בעל ארבעה מנועים

מאת:

חנוך אפרים

מנחה: דר. אמיר שפירא

חיבור זה מהווה חלק מהדרישות
לקבלת התואר "מגיסטר" בהנדסה

אוקטובר 2011

תשרי תשע"ב